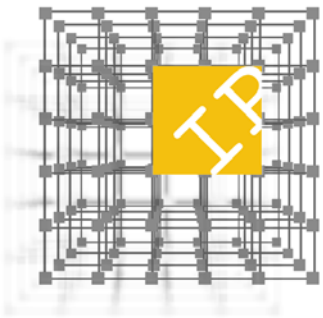
PDnet™



PAD-250e Benutzerhandbuch

IPEX
automation technologies GmbH

Einschränkung der Gewährleistung:

Die Angaben und Hinweise in diesem Handbuch sind sorgfältig geprüft und entsprechen den verfügbaren Daten zur Zeit der Drucklegung. Es wird keine Garantie für die Richtigkeit des Inhaltes übernommen.

Die im Handbuch verwendeten Software- und Hardwarebezeichnungen sind zum Teil eingetragene Warenzeichen und unterliegen als solche den gesetzlichen Bestimmungen.

Herausgeber:

APEX automation technologies GmbH
Vossenkamp 4
38104 Braunschweig

Telefon 0531-3704-0
Telefax 0531-3704-299

http://www.apex.de

CE

Der PDnetIP-Controller erfüllt die Anforderungen der EU-Richtlinie 89/336/EWG „Elektromagnetische Verträglichkeit“. Die Konformität des PDnet-Controllers mit der o.g. Richtlinie wird durch das CE-Zeichen bestätigt.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

INHALT

INHALT 3

1 Einleitung 5
1.1 Produktübersicht 6
1.2 Bestimmungsgemäße Verwendung 8
1.3 Allgemeine Hinweise 9
1.4 Lieferumfang 9

2 Technische Daten 10
2.1 Hardware 11
2.1.1 Ethernet 11
2.1.2 Serielle Schnittstellen 12
2.2 Software 13
2.2.1 Leistungsmerkmale 13
2.2.2 TCP/IP 14
2.2.2.1 ICOS/TCP 15
2.2.2.2 ModBus/TCP 17
2.2.2.3 RFC1006 (ISO TP0 over TCP) 19
2.2.2.4 OpenTCP 25
2.2.2.5 MSI-Schnittstelle 34
2.2.2.6 Kundenspezifische Treiber 42
2.2.3 Serielle Protokolle 43
2.2.3.1 Access-Server 43
2.2.3.2 IS-Tester 43
2.2.3.3 ModBus Master 47
2.2.3.4 ModBus Slave 47
2.2.3.5 Seab 1/F Master 48
2.2.3.6 Kundenspezifische Treiber 51

3 Installation des PAD-250e 52
3.1 Einbau des PAD-250e 52
3.2 Sonder-Funktions-Bausteine (SFB) 52
3.2.1 SFB693 - PDNIP 53
3.2.2 SFB692 - PDSTR 56
3.2.3 SFB695 – POILE 58
3.2.4 SFB696 – PDSTI 59
3.2.5 SFB691 - PVDMS 61
3.2.6 Integration der SFB 62

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

0 INHALT
1.1 Produktübersicht

3.2.6.1 AKF125 ... 62
3.2.6.2 ALD25 für OS/2 ... 65
3.2.6.3 ALD25 für Windows NT/XP ... 65
3.2.7 Verfügbarkeit ... 66
3.3 AKF/ALD Bestückungsliste ... 67
3.4 Strukturen ... 67
3.5 ICOS Projektierung mit COMAKF ... 68

4 Anwendung ... 69
4.1 ICOS/TCP ... 69
4.1.1 ICOS-Diagnose Modul ... 69
4.2 ModBus/TCP ... 71
4.2.1 Strukturen ... 72
4.2.2 ModBus-Auftragsblock ... 73
4.2.3 Beispiele ... 78
4.2.3.1 Read Register ... 78
4.2.3.2 Write Register ... 79
4.2.3.3 Read Coil ... 79
4.2.3.4 Write Coil ... 80
4.3 ISO TP0 over TCP (RFC1006) ... 81
4.3.1 Projektierung mit NetPro ... 82
4.4 OpenTCP-Schnittstelle ... 87

5 Anzeige- und Bedienelemente ... 88
5.1 Status-Leuchtdioden ... 88
5.1.1 Ethernet Status ... 89
5.1.2 Firmware Status ... 90
5.1.3 Serielle Schnittstellen ... 91
5.2 Taster ... 92
5.2.1 Lader aktivieren ... 93
5.2.2 Lader verlassen ... 94

6 Konfiguration ... 95
6.1 Projektierung ... 95
6.2 Firmware ... 96
6.3 Lader ... 97
6.3.1 Lader-Versionen ... 98
6.3.2 Lader-Aktualisierung ... 99

7 Anhang ... 100
7.1 Fehlerliste TCP/IP-Stack ... 100

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

1 Einleitung

Dieses Handbuch unterstützt Sie beim Einbau, der Konfiguration und der Inbetriebnahme des PDnetIP-Controllers PAD-250e. Es beschreibt die Installation folgender Produktvarianten:

- **PAD-250e** verfügbar als
  PAD-250e/T, PAD-250e/L

Die APEX automation technologies GmbH behält sich Änderungen und Weiterentwicklungen vor, die zu Abweichungen von den in diesem Handbuch angegebenen Daten führen können.

Die Hardware wird von der APEX automation technologies GmbH zusammen mit einer PDnetIP-CD geliefert welche die für den Betrieb des PDnetIP-Controllers notwendige Software und Dokumentation beinhaltet.

Bestandteil der PDnetIP-CD ab Version 2.41 ist das Dokument „PDnetIP Versionsübersicht“. Das Dokument gibt einen Überblick über die einzelnen Versionen.

Benutzen Sie die Software und Dokumentation einer PDnetIP-CD’s immer als Einheit.

Die aktuelle PDnetIP-CD kann unter der Adresse ftp2.apex.de abgerufen werden. Benutzen Sie für den Download den Benutzernamen **`kunde`** und das Passwort **`apex`**.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

1.1 Produktübersicht

Der PDnetIP-Controller PAD-250e ist ein intelligenter PAB1-Teilnehmer. Mit der ICOS-Version des PAD-250e können die Modnet-Koppler vom Typ BIK-114, KPH-141 und KPM-111 ersetzt werden und A250-Automatisierungsgeräte an das PDnetIP (Ethernet, TCP/IP) angeschlossen werden.

Übersicht der PAD-250e Varianten

Artikel Nummer	Name	Beschreibung
20102050	PAD-250e/T	PDnetIP-Controller für Modicon TSX A250, mit RJ 45-Buchse
20102051	PAD-250e/T	PDnetIP-Controller für Modicon TSX A250, mit RJ 45-Buchse (inkl. ICOS)
20102052	PAD-250e/L	PDnetIP-Controller für Modicon TSX A250, mit RJ 45-Buchse und MT-RJ-Buchse
20102053	PAD-250e/L	PDnetIP-Controller für Modicon TSX A250, mit RJ 45-Buchse und MT-RJ-Buchse (inkl. ICOS)

Der PDnetIP-Controller ist in verschiedenen Versionen erhältlich. Aus lizenzrechtlichen Gründen darf ICOS nicht in jeder Version zur Verfügung stehen. Ergänzend zu den in der oberen Tabelle dargestellten Versionen, gibt es weitere optional erhältliche Module. Die folgende Tabelle gibt eine Übersicht welche Protokolle lizenzrechtlich wann zur Verfügung stehen. Ob die gewünschte Kombination von Protokollen gleichzeitig nutzbar sind, ist separat zu prüfen.

Protokolle	Anmerkung
ModBus/TCP	Steht in jedem PDnetIP-Controller zur Verfügung.
VDM (PDnetIP)	
ICOS	Steht nur in den ICOS-Versionen zur Verfügung. Artikel-Nummern: 20102051 und 20102053
RFC1006	Separat zu erwerbende Lizenz
OpenTCP	Separat zu erwerbende Lizenz

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Der PDnetIP-Controller ist mit wechselbaren Leitungsinterface-modulen (LIM) bestückt, welche die Leitungsart der seriellen Schnittstellen anpassen.

Die Lieferung erfolgt mit bestücktem LIM-232. Die Module können durch den Anwender auf dem PDnetIP-Controller gewechselt werden.

Übersicht der Leitungsinterfacemodule

Artikel Nummer	Name	Beschreibung
10213010	LIM-232	LIM-232 Leitungs-Interface-Modul für RS-232 ohne Potentialtrennung
10213011	LIM-422	Leitungs-Interface-Modul für RS-422 ohne Potentialtrennung
10213012	LIM-485	Leitungs-Interface-Modul für RS-485 ohne Potentialtrennung
10213016	LIM-20 mA	Leitungs-Interface-Modul für 20 mA Stromschleife mit Potentialtrennung

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

1.2 Bestimmungsgemäße Verwendung

Der PDnetIP-Controller PAD-250e ist ausschließlich dazu bestimmt:

- eine MODICON TSX A250 mit dem PDnetIP zu verbinden.

Wird der PAD-250e anders als in diesem Handbuch beschrieben verwendet, ist kein sicherer Betrieb gewährleistet.

Für Schäden, Mehraufwendungen, Probleme, Störungen und insbesondere Personen- und Sachschäden, die aus nicht bestimmungsgemäßer Verwendung entstehen, ist nicht der Hersteller verantwortlich.

 © 1992-2014 by APEX® automation technologies GmbH, Braunschweig

1.3 Allgemeine Hinweise

Der Einbau des PDnetIP-Controllers sollte von einer Fachperson durchgeführt werden. Die APEX automation technologies GmbH trägt keine Verantwortung für Schäden, die durch einen unsachgemäßen Einbau entstehen und/oder entstanden sind.

Das vorliegende Benutzerhandbuch erläutert den Einbau und die Inbetriebnahme der PDnetIP-Controller. Informationen über den Umgang mit dem Computersystem bzw. dem (den) Betriebssystem(en), die im Zusammenhang mit diesem Computersystem betrieben werden sollen, sind den entsprechenden System-Handbüchern zu entnehmen.

Nachdem die Verpackung geöffnet und dieses Handbuch, sowie der PDnetIP-Controller herausgenommen wurden, sollte der PDnetIP-Controller bis zu seinem Einbau in dem Antistatik-Schutzbeutel verbleiben.

1.4 Lieferumfang

Zum Lieferumfang der PAD-250e gehören:

1. PAD-250e in Antistatikverpackung.
2. CD-ROM PDnetIP.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

2 Technische Daten

Zuordnung	
Zentralbaugruppen	ALU 151, ALU 152, ALU 153, ALU 154
Anschluss	PAB1-Schnittstelle über DPM (Dual-Port-Memory). Bis einschließlich HW-Revision 3, Index 1 über 8 KB DPM und ab Hardware-Revision 3, Index 2 bis zu 64 KB DPM. Die Größe des zur Kommunikation nutzbaren DPM hängt von der Konfiguration und der einsetzten Software ab.
Versorgung	
Betriebsspannung	5 V (+/- 5%) / 900 mA von PAB1-Schnittstelle
PDnetIP-Anschlüsse	
Anzahl	1
Übertragungsgeschwindigkeit	10/100 Mbit/Sekunde Ethernet 10Base-T, 100Base-TX, 100Base-FX (LWL)
Kabeltyp	Twisted-Pair mit RJ-45 Stecker Cat. 5 LWL mit MT-RJ Anschluss
Serielle Schnittstellen	
Anzahl	2
Leitungsphysik	Wählbar über Steckmodul: RS-232, RS-485, RS-422, 20 mA
Übertragungsgeschwindigkeit.	50 ─ 115.200 Bit/Sekunde (die nutzbaren Baudraten sind in den Protokollen verschieden)
Anschluss	9-polige SubD-Buchse
Prozessor	
Typ	Intel 386EX
Speicher	Bis Hardware-Revision 2: 1MB SRAM und 1MB Flash-memory und ab Hardware-Revision 3: 2MB SRAM und 2MB Flashmemory
Mechanischer Aufbau	
Kartenformat	Doppeleuropakarte (234 mm x 160 mm)
Breite	4 TE
Gewicht	350 g
Umgebungsbedingungen	
Temperatur	0 - 60 °C (im Betrieb), -40 - +85 °C (Lagerung)
Luftfeuchtigkeit	10 % ─ 80 % (nicht kondensierend)

Es gibt keine Einschränkungen wie viele PAD-250e gleichzeitig in einem Magazin betrieben werden können.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

2.1 Hardware

2.1.1 Ethernet

Der PAD-250e/L ist mit zwei Schnittstellen ausgestattet. Er besitzt sowohl einen RJ45-Anschluss als auch ein MT-RJ-Anschluss. Zu einem Zeitpunkt kann nur ein Anschluss genutzt werden. **Wird der PDnetIP-Controller über den MT-RJ-Anschluss betrieben, so muss auf der Gegenstelle als Betriebsart Halb-Duplex konfiguriert werden.** Die Mac-Adresse (Ethernet-Adresse) des PDnetIP-Controllers ist in der Hardware gespeichert und unveränderlich. Alle gelieferten PDnetIP-Controller nutzen eine mit **`00:05:DA`** beginnende Mac-Adresse.

Wichtig: Es wird empfohlen einen über LWL angeschlossenen PAD-250e im Halbduplex-Mode zu betreiben. Im Vollduplex-Mode gibt es Kollisionszustände, welche nicht erkannt werden können, was zu Telegramm-Wiederholungen führt. Die Ursache liegt darin, dass auch in einem geswitchten Ethernet Kollisionen entstehen können.

Die folgende Tabelle zeigt die Belegung der RJ-45 Anschlussbuchse, die der IEEE802.3 Twisted Pair-Schnittstelle entspricht.

Pin	Signal
1	Transmit +
2	Transmit -
3	Receive +
4	-
5	-
6	Receive -
7	-
8	-

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

2.1.2 Serielle Schnittstellen

Der PAD-250e ist mit 2 seriellen Schnittstellen ausgestattet. Die Schnittstellen sind als 9-polige SubD-Buchsen auf der Frontseite ausgeführt. Die linke SubD-Buchse ist die erste serielle Schnittstelle und entsprechend die rechte die zweite serielle Schnittstelle.

Die Leitungsinterfacemodule (LIM-Module) passen die seriellen Schnittstellen des PDnetIP-Controllers an die Leitungsphysik an. Die Standardausstattung erfolgt mit bestücktem LIM-232.

Die LIM-Module können durch den Anwender auf dem PDnetIP-Controller gewechselt werden.

Die folgende Tabelle zeigt die PIN-Belegung der seriellen Schnittstellen in Abhängigkeit der eingesetzten LIM-Module.

PIN Belegung der seriellen Schnittstellen

Pin	LIM-232 V.24 Signal	LIM 485 RS 485	LIM-422 RS 422	LIM-20 mA Stromschleife
1	DCD	–	–	Tx-
2	RxD	TxD +	TxD +	Rx +
3	TxD	RxD +	RxD +	Tx +
4	DTR	+5V	+5V	–
5	GND	GND	GND	GND
6	–	–	–	Rx -
7	RTS	RxD-	RxD-	RTS (TTL)
8	CTS	TxD-	TxD-	CTS (TTL)
9	–	–	–	–

– nicht belegt

Die folgende Tabelle zeigt, wie ein Programmierkabel zu konfektionieren ist, um den PDnetIP-Controller über die serielle Schnittstelle mit dem Programm DLSE32 laden zu können. Alternativ kann das Standard-A250 Programmierkabel YDL52 genutzt werden.

PDnetIP-Controller	Programmiergerät (PC, 9poliger Anschluss)
2	3
3	2
5	5

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

2.2 Software

Der Leistungsumfang der PDnetIP-Controller wird im wesentlichen von der verwendeten Firmware bestimmt. Wird die Firmware auf dem PDnetIP-Controller getauscht ist das entsprechend für diese Firmware gültige Handbuch zu nutzen.

Um die Dienste ICOS, RFC1006 oder die OpenTCP nutzen zu können, müssen diese auf der Karte freigeschaltet sein. Der nachfolgende Text geht immer von der Verfügbarkeit der entsprechenden Dienste aus.

2.2.1 Leistungsmerkmale

Protokoll	Limit – Maximum
ARP	64 ARP-Einträge
ICOS/TCP	32 Verbindungen
ModBus/TCP	32 Verbindungen
RFC1006	50 Verbindungen
OpenTCP	50 Verbindungen
MSI-Schnittstelle	5 Verbindungen

Folgende Limits gelten für alle Protokolle in Summe

- 128 TCP-Verbindungen
- 5 UDP-Verbindungen

Eine Reserve von 10% sollte berücksichtigt werden.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

2.2.2 TCP/IP

Der PDnetIP-Controller nutzt die in der Tabelle aufgeführten Port-Nummern. Diese Informationen sind für den Netzwerk-Administrator von Bedeutung, wenn aktive Netzwerkkomponenten wie Router oder Firewalls genutzt werden.

Protokoll	Port-Nummer
DNS	TCP 53
BOOTP/DHCP	UDP 68
ISO TP0 over TCP (RFC1006)	TCP 102
ModBus/TCP	TCP 502
MSI	Abhängig von der Projektierung
PDnetIP (VDM)	UDP/TCP 2843
ICOS/TCP	UDP/TCP 9876
TCP	Abhängig von der Projektierung
SYSLOG	UDP 514
SNTP	UDP 123

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

2.2.2.1 ICOS/TCP

Der PDnetIP-Controller ist in der Lage, ICOS-Telegramme über TCP/IP zu transportieren und so ältere Netzwerke, wie Modnet-2NP, 2ND, ICOS via H1 u.ä., zu substituieren. In diesem Fall läuft die Funktionalität des alten Netzes als Dienst auf dem TCP/IP-Netz mit. Zu diesem Zweck stellt der PDnetIP-Controller der SPS ein ihr „bekanntes“ Interface bereit. Die Steuerung bemerkt keinen Unterschied und wickelt die Kommunikation wie gewohnt ab. Dies ermöglicht einen nahtlosen Übergang zu neuen Lösungen. Dabei erweitert der PDnetIP-Controller das herkömmliche Interface um weitere Leistungsmerkmale.

Als Interface kann ein KP-Interface (KPH-141 bzw. KPM-111) oder BIK-114 genutzt werden. Die Betriebsart mit ICOS/TCP wird auch benötigt, um eine A250 über TCP/IP bedienen und laden zu können. Folgende ALD-Versionen unterstützen diese Funktionen:

- ALD, Version 9 für Windows NT/2000
- ALD25XP, Version 10 für Windows NT/2000/XP
- ALD25XP, Version 10.5 für Windows NT/2000/XP

Es werden alle ICOS-Telegrammtypen inkl. B8EFELD ohne Einschränkungen hinsichtlich der Telegrammlänge unterstützt. Der PAD-250e muss nicht in die Bestückungsliste der A250 eingetragen werden um ICOS-Dienste nutzen zu können.

Der PDnetIP-Controller kann zu einem Zeitpunkt nur eine limitierte Anzahl von ICOS-Verbindungen zu verschiedenen Partnerstationen aufrecht halten. Sind über eine IP-Adresse verschiedene A-Bytes erreichbar, zählt dies als eine Verbindung.

Wichtig: Es müssen alle A-Bytes projektiert werden mit welchen der PDnetIP-Controller kommunizieren soll. Dies trifft auch auf PC-Systeme zu, welche als externe Station definiert werden. Nur so kann der PDnetIP-Controller ein A-Byte in eine IP-Adresse bzw. einen DNS-Namen auflösen.

Die Projektierung darf ausschließlich mit statischem A-Byte erfolgen!

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

2 Technische Daten
2.2 Software

Die PDnetIP-Controller unterstützen die Verarbeitung folgender ICOS Prozeduren.

Prozedur	A-Byte	Broadcast-Adresse
BUS_1_MA	240	127
BUS_1_SL	1	
BUS_2_TA	wie projektiert	
BUS_2_TB		255
BUS_2_FM		
STERN_1_MA	240	127
STERN_1_SL	1	

Der PDnetIP-Controller unterstützt BUS- und STERN-Prozeduren, welche ursprünglich nur zur seriellen Kommunikation vorgesehen waren. Bei diesen Prozeduren gelten die in der Tabelle dargestellten Sonderregeln zur Bestimmung des A-Byte. Damit ist es dem PDnetIP-Controller möglich, die Prozeduren ohne Anpassungen auf der SPS-Seite zu unterstützen.

Wichtig! Alle BroadCast-Telegramme werden an die BroadCast-Adresse des Subnetzes gesendet. Damit können die Stationen, welche Mitglied des Subnetzes sind, BroadCast-Telegramme empfangen.

Wichtig! Wird die ICOS-Kommunikation von der SPS initialisiert, (Einschaltmoment oder Menü Remote-Kopplung initialisieren) übergibt die SPS das A-Byte an den PDnetIP-Controller. Das A-Byte wird vom PDnetIP-Controller mit dem projektierten A-Byte verglichen und nur bei einer Übereinstimmung die Initialisierung quittiert.

Zur erweiterten Diagnose steht ein optionales ICOS-Diagnose-Modul zur Verfügung, welches dem Anwenderprogramm der SPS einen Überblick gibt, zu welcher ICOS-Station (A-Byte) eine Verbindung vorhanden ist. Konnte keine Verbindung aufgebaut werden oder ist die Verbindung gestört, werden der SPS Informationen zur Ursache der Störung zugänglich gemacht.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

2.2.2.2 ModBus/TCP

Der PDnetIP-Controller unterstützt ModBus/TCP Funktionen im Server- und Master-Betrieb. Dazu benötigt der PDnetIP-Controller mit Hilfe des PDNIP-Bausteins Zugriff auf den Signalspeicher der Steuerung.

Der PDnetIP-Controller kann zu einem Zeitpunkt nur eine limitierte Anzahl von ModBus/TCP-Verbindungen aufrecht halten. Eine Station (IP-Adresse) kann gleichzeitig mehrere Verbindungen zu den PDnetIP-Controllern unterhalten und dadurch eine bessere Kommunikationsgeschwindigkeit erreichen.

Nachfolgende Tabelle gibt einen Überblick der unterstützten ModBus/TCP-Funktionen:

- 01 (0x01) Read Coils
- 02 (0x02) Read Discrete Input
- 03 (0x03) Read Holding Register
- 04 (0x04) Read Input Register
- 05 (0x05) Write Single Coil
- 06 (0x06) Write Single Register
- 07 (0x07) Read Exception State
- 15 (0x0F) Write Multiple Coils
- 16 (0x10) Write Multiple Registers

Über ModBus/TCP können folgende Datentypen der A250 adressiert werden.

- Merker
- Merkerbyte
- Merkerworte
- Worte
- Doppelworte
- Gleitkommaworte

Für jeden Datenbereich muss zuvor mit NetPro ein Mapping definiert werden.

Ein Mapping ordnet den ModBus/TCP Datentypen Register und Coils einem Speicherbereich der SPS zu. Dabei wird festgelegt, wie die Datentypen auf einander abgebildet werden und wie viele

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Elemente dieser Bereich umfasst. Zusätzlich kann jeder Bereich mit Eigenschaften wie "Read Only" oder Informationen zur Byte-Order (Word-Swap) konfiguriert werden. Für einen ModBus/TCP Dienst können mehrere verschiedene dieser Bereiche angelegt werden. Eine ModBus/TCP Transaktion, die sich auf Elemente bezieht, die nicht vollständig einem Mapping zugeordnet werden können, wird zurückgewiesen.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

2.2.2.3 RFC1006 (ISO TP0 over TCP)

Der PDnetIP-Controller unterstützt optional RFC1006 bzw. „ISO TP0 over TCP“. Dazu benötigt er mit Hilfe des PDNIP-Bausteins Zugriff auf den Signalspeicher der Steuerung.

Auf Netzebene werden die Daten gemäß ISO 8073 (TP0) über ein TCP/IP Netzwerk in Blöcken übertragen. Der PDnetIP-Controller überträgt die Netto-Daten direkt in den Signalspeicher der Steuerung.

Die RFC1006-Projektierungsinformationen werden als eine Liste von Verbindungen bearbeitet. Jede Verbindung besitzt folgende Eigenschaften:

- Eine Kennung (1..65535) sowie einen symbolischen Namen (24 Zeichen) zur eindeutigen Identifikation der Verbindung.
- Die IP-Adresse der Partnerstation, den eigenen TSAP sowie den TSAP der Partnerstation. Der TSAP kann eine Länge von 2 bis 16 Zeichen besitzen.
- Die genutzte Auftragsart (SEND, RECEIVE oder SEND und RECEIVE) sowie optionale Konfigurationsparameter. Eine Verbindung mit der Auftragsart SEND kann nur Daten senden und keine Daten empfangen. Entsprechend kann eine Verbindung mit der Auftragsart RECEIVE nur Daten empfangen und keine Daten senden. Die Betriebsart SEND/RECEIVE ermöglicht eine bidirektionale Kommunikation.
- Die Adresslage des Datenpuffers im Signalspeicher der Steuerung. Die Adresslage wird dabei für Strukturen als Kennung (symbolisch) und für andere Adressbereiche absolut als Signalspeicheradresse definiert.

Jede projektierte Verbindung belegt im Signalspeicher der Steuerung vier Merker, sowie vier Worte. Über diese Daten kommuniziert das Anwenderprogramm der SPS mit dem RFC1006-Treiber der Firmware.

Zur Ablage dieser Informationen ist vorab eine Startadresse im Wort- sowie eine Startadresse im Merkerbereich der Steuerung zu definieren. Die Adressen werden vom RFC1006-Treiber automatisch belegt.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

NetPro exportiert die Belegung der Signalspeicheradressen als ASD-Datei. Die so exportierten Informationen können direkt mit AKF oder ALD eingelesen werden.

Der Symbolname setzt sich aus dem Verbindungsnamen sowie einem Suffix zusammen. Im Anwenderprogramm der SPS kann über die symbolischen Namen einfach auf die Informationen zugegriffen werden.

Nach jeder Veränderung der RFC1006 Projektierung (Verbindung einfügen oder löschen) kann sich die Belegung des Signalspeichers verändern.

Die folgenden Tabellen beschreiben die Bedeutung der für jede Verbindung belegten Status-Worte und Merker. Der Suffix ergänzt den mit NetPro projektierten Verbindungsnamen.

Suffix	Vorhanden	Zustand
'_ID'	Immer	Die mit NetPro projektierte Verbindungskennung.
'_STATE'		Der aktuelle Verbindungsstatus.
'_SB'	Senderichtung	Beauftragungsbit in Senderichtung. Über das Sendebit zeigt das Anwenderprogramm der SPS an, dass ein Sendetelegramm vorliegt.
'_SQ'		
'_AWS'		Letzter/Aktueller Status in Senderichtung.
'_EB'	Empfangsrichtung	Das Empfangsbit informiert das Anwenderprogramm der SPS über ein neues Empfangstelegramm.
'_EQ'		
'_AWE'		Letzter/Aktueller Status in Empfangsrichtung.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Mögliche Zustände SB/SQ

SB	SQ	Zustand
0	1	Sendepuffer frei. Dieser Zustand wird vom PDnetIP-Controller im Einschaltmoment gesetzt, so dass im Anwenderprogramm der SPS keine Normierung der SB/SQ-Merker durchgeführt werden muss.
1	0	Wird vom Anwenderprogramm der SPS gesetzt, wenn ein Telegramm gesendet werden soll. Wurde der Sendevorgang angestoßen, darf dieser vom Anwenderprogramm der SPS nicht wieder gelöscht werden. Das Anwendungsprogramm der SPS hat die Kontrolle an den PDnetIP-Controller abgegeben und muss passiv auf die Reaktion warten. **Veränderung von SB/SQ ist nicht zulässig.**
0	0	Zustand ist nicht zugelassen und wird vom PDnetIP-Controller auf SB=0 und SQ=1 normiert.
1	1	

Ablauf Sendevorgang

Schritt	SB	SQ	Zustand
1	0	1	Der Sendepuffer ist verfügbar.
2	1	0	Das Anwenderprogramm der SPS hat den Sendepuffer mit Daten befüllt und zeigt über das gesetzte Sendebit den Sendewunsch an.
3	0	1	Der PDnetIP-Controller hat den Sendewunsch bearbeitet und der Sendepuffer steht wieder zur Verfügung.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Mögliche Zustände EB/EQ

EB	EQ	Zustand
0	0	Der Empfangspuffer ist frei. Dieser Zustand wird vom PDnetIP-Controller im Einschaltmoment gesetzt, so dass im Anwenderprogramm der SPS keine Normierung der EB/EQ-Merker durchgeführt werden muss.
0	1	Zustand ist nicht zugelassen und wird vom PDnetIP-Controller normiert.
1	0	Der PDnetIP-Controller hat ein Telegramm empfangen und im Empfangspuffer abgelegt. Das Telegramm muss vom Anwenderprogramm der SPS quittiert werden. Erfolgt dies nicht (SPS wird angehalten), wird das Telegramm über ein TimeOut gelöscht.
1	1	Das Anwenderprogramm der SPS hat das Telegramm verarbeitet und zeigt dies über das Setzen von EQ dem PDnetIP-Controller an.

Ablauf Empfangsvorgang

Schritt	EB	EQ	Zustand
1	0	0	Empfangspuffer ist frei
2	1	0	Es wurde ein Telegramm empfangen
3	1	1	Das Anwenderprogramm der SPS hat das Telegramm verarbeitet
4	0	0	Empfangspuffer ist wieder frei

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Fehlerliste Ausgangsworte (_AWS und _AWE)

Wert	Bedeutung
0	Kein Fehler
100	Für die TSAP-Kombination wurde eine bereits vorhandene Verbindung durch einen neue eingehende Verbindung geschlossen.
102	Die Verbindung wurde durch einen Timeout geschlossen.
104	Verbindung wurde geschlossen
105	Aktiver Verbindungsaufbau gescheitert
106	Keine Verbindung vorhanden – Sendedaten können nicht zugestellt werden, da der PDnetIP-Controller selbst keine Verbindung aufbauen darf (Kann in der Konfiguration über „Aktiver Verbindungsaufbau“ geändert werden)
107	Sendevorgang gescheitert
109	Die Zahl gleichzeitig ausführbarer Sendevorgänge von 12 PDU´s pro Verbindung wurde überschritten.
111	TimeOut Sendedaten
112	TimeOut Empfangsdaten
113	Empfangsdaten konnten nicht verarbeitet werden, da die Struktur der Empfangsdaten zuvor nicht registriert wurde. Strukturen müssen über den Bausteine PDSTR (SFB692) beim PDnetIP-Controller registriert werden.
114	Sendedaten konnten nicht verarbeitet werden da die Struktur der Sendedaten zuvor nicht registriert wurde. Strukturen werden mit dem Baustein PDSTR (SFB692) beim PDnetIP-Controller registriert.
115	Der empfangene „Connection Request“ konnte nicht verarbeitet werden.
116	Die Verbindung wurde geschlossen
117	Fehlerhafte RFC1006-Daten empfangen

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Verbindungsstatus (_STATE)

Wert	Bedeutung
1	Verbindung wurde von der Partner-Station geöffnet. Dieser Status lieg vor, wenn die Verbindung im PDnetIP-Controller als passiv projektiert wurde.
2	Der PDnetIP-Controller baut eine Verbindung auf. Der Status steht solange an, bis der eigene CR mit einem CC der Partnerstation bestätigt wurde.
3	Der PDnetIP-Controller besitzt eine aktive Verbindung zur Partnerstation. Diese Verbindung wurde aktiv aufgebaut.
4	Es besteht keine Verbindung zu einer Partnerstation.

Die maximale Größe der TPDU´s beträgt 2.048 Byte. Baut der PDnetIP-Controller aktiv eine Verbindung auf, so wird im „Connection Request“ eine TPDU-Größe von 2.048 Byte festgelegt. Antwortet die Partnerstation im „Connection Confirm“ mit einer kleineren TPDU-Größe, nutzt der PDnetIP-Controller diese.

Baut eine andere Station aktiv eine Verbindung zum PDnetIP-Controller auf, nutzt dieser die im „Connection Request“ übermittelte TPDU-Größe, so diese nicht größer als 2.048 Byte ist. Andernfalls quittiert der PDnetIP-Controller im „Connection Confirm“ die TPDU-Größe mit 2.048 Byte und erwartet, dass die Partnerstation diese Größe in den Sendedaten benutzt.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

2.2.2.4 OpenTCP

Der PDnetIP-Controller unterstützt **optional** eine freie OpenTCP-Schnittstelle. Dazu benötigt er mit Hilfe des PDNIP-Bausteins Zugriff auf den Signalspeicher der Steuerung.

Auf Netzebene werden dazu TCP-Daten direkt in den Signalspeicher der Steuerung übertragen.

Die zur Verarbeitung notwendigen Informationen werden vorab mit NetPro projektiert, und im nichtflüchtigen Setupdatenspeicher des PDnetIP-Controllers abgelegt.

Die TCP-Projektierungsinformationen werden als eine Liste von Verbindungen erstellt. Jede Verbindung besitzt folgende Eigenschaften:

- Eine Kennung (1..65535) zur eindeutigen Identifikation der Verbindung.
- Ein Name (24 Zeichen) zur symbolischen Identifikation der Verbindung.
- Die IP-Adresse der Partnerstation sowie die zugeordnete TCP-Portnummer.
- Die genutzte Auftragsart (SEND, RECEIVE oder SEND und RECEIVE). Eine Verbindung mit der Auftragsart SEND kann nur Daten senden und keine Daten empfangen. Entsprechend kann eine Verbindung mit der Auftragsart RECEIVE nur Daten empfangen und keine Daten senden. Die Betriebsart SEND/RECEIVE ermöglicht eine bidirektionale Kommunikation.
- Die optionale Übertragungsart „Header mit Kennung“, die vor den Nettodaten einen Header sendet, welcher es der Partnerstation ermöglicht die Daten zu unterscheiden.
- Die Adresslage des Datenpuffers im Signalspeicher der Steuerung. Die Adresslage wird dabei für Strukturen als Kennung (symbolisch) und andernfalls im Wortbereich der Steuerung definiert. Wird die Option „Header mit Kennung“ genutzt wird eine Liste der Datenpuffer definiert.

Jede projektierte Verbindung belegt im Signalspeicher der Steuerung vier Merker, sowie acht Worte. Über diese Daten kommuniziert das Anwenderprogramm der SPS mit der OpenTCP-Schnittstelle der Firmware.

Zur Ablage dieser Informationen ist vorab eine Startadresse im Wort- sowie eine Startadresse im Merkerbereich der Steuerung zu definieren. Die Adressen werden von der OpenTCP-Schnittstelle automatisch belegt.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

NetPro exportiert die Belegung der Signalspeicheradressen als ASD-Datei. Die so exportierten Informationen können direkt mit AKF oder ALD eingelesen werden.

Jeder in eine ASD-Datei exportierte Symbolname setzt sich aus dem Verbindungsnamen, sowie einem Suffix zusammen. Im Anwenderprogramm der SPS kann über die symbolischen Namen einfach auf die Informationen zugegriffen werden.

Nach jeder Veränderung der OpenTCP-Schnittstellen-Projektierung (Verbindung einfügen oder löschen), kann sich die Belegung des Signalspeichers verändern.

Die folgende Tabelle beschreibt die Bedeutung der für jede Verbindung belegten Status-Worte und Merker. Der Suffix ergänzt den mit NetPro projektierten Verbindungsnamen.

Suffix	Vorhanden	Zustand
'_ID'	Immer	Die mit NetPro projektierte Verbindungskennung.
'_STATE'		Der aktuelle Verbindungsstatus.
'_SB'	Sende-richtung	Beauftragungsbit in Senderichtung. Über das Sendebit zeigt das Anwenderprogramm der SPS, dass ein Sendetelegramm vorliegt.
'_SQ'		
'_TXSTATE'		Letzter/Aktueller Status in Senderichtung.
,_TXLEN'		Anzahl zu sendender Bytes
,_TXSTRUC'		Kennung der zu sendenden Struktur *)
'_EB'	Empfangs-richtung	Das Empfangsbit informiert das Anwenderprogramm der SPS über ein neues Empfangstelegramm.
'_EQ'		
'_RXSTATE'		Letzter/Aktueller Status in Empfangsrichtung.
,_RXLEN'		Anzahl empfangener Bytes
,_RXSTRUC'		Kennung der empfangenen Struktur *)

*) Diese Worte werden nur in der Übertragungsart „Header mit Kennung" genutzt, in diesem Fall bezieht der PDnetIP-Controller die Längeninformationen weder aus RXLEN und TXLEN noch aus den projektierten Strukturen.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Die nachfolgende Legende zeigt die im folgenden genutzte farbliche Kodierung der Seite, die für das Setzen des Zustandes verantwortlich ist.

Der Zustand wird vom PDnetIP-Controller gesetzt und dient dem Anwenderprogramm der SPS als Triggerbedingung.

Der Zustand wird vom Anwenderprogramm der SPS gesetzt und dient der Firmware des PDnetIP-Controllers als Triggerbedingung.

Der Zustand ist nicht zulässig und wird von einem fehlerhaften Anwenderprogramm der SPS verursacht.

Mögliche Zustände SB/SQ

SB	SQ	Zustand
0	1	Sendepuffer frei. Dieser Zustand wird vom PDnetIP-Controller auch im Einschaltmoment gesetzt, so dass im Anwenderprogramm der SPS keine Normierung der SB/SQ-Merker durchgeführt werden muss.
1	0	Wird vom Anwenderprogramm der SPS gesetzt, wenn ein Telegramm gesendet werden soll. Wurde der Sendevorgang angestoßen, darf dieser vom Anwenderprogramm der SPS nicht wieder gelöscht werden. In jedem Fall muss das Anwenderprogramm der SPS auf die Reaktion des PDnetIP-Controllers warten.
0	0	Zustand ist nicht zugelassen und wird vom PDnetIP-Controller auf SB=0 und SQ=1 normiert.
1	1	

Ablauf Sendevorgang

Schritt	SB	SQ	Zustand
1	0	1	Der Sendepuffer ist verfügbar. Der Zustand wird von dem PDnetIP-Controller initialisiert. Auf diesen Zustand muss das Anwenderprogramm der SPS warten, um einen Sendevorgang starten zu können.
2	1	0	Das Anwenderprogramm der SPS hat den Sendepuffer mit Daten gefüllt und startet so den Sendevorgang.
3	0	1	Der PDnetIP-Controller hat den Sendewunsch bearbeitet und der Sendepuffer steht wieder zur Verfügung. Ob der Sendevorgang erfolgreich war, kann TXSTATE entnommen werden.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Mögliche Zustände EB/EQ

EB	EQ	Zustand
0	0	Empfangspuffer frei. Dieser Zustand wird vom PDnetIP-Controller im Einschaltmoment gesetzt, so dass im Anwenderprogramm der SPS keine Normierung der EB/EQ-Merker durchgeführt werden muss.
0	1	Zustand ist nicht zugelassen und wird vom PDnetIP-Controller normiert.
1	0	Der PDnetIP-Controller hat ein Telegramm empfangen und im Empfangspuffer abgelegt. Das Telegramm muss vom Anwenderprogramm der SPS durch Setzen des EQ-Bit quittiert werden. Erfolgt dies nicht (SPS wird angehalten), wird das Telegramm über ein TimeOut gelöscht.
1	1	Das Anwenderprogramm der SPS hat das Telegramm verarbeitet und quittiert.

Ablauf Empfangsvorgang

Schritt	EB	EQ	Zustand
1	0	0	Empfangspuffer frei. Dieser Zustand wird vom PDnetIP-Controller im Einschaltmoment gesetzt, so dass im Anwenderprogramm der SPS keine Normierung der EB/EQ-Merker durchgeführt werden muss.
2	1	0	Der PDnetIP-Controller hat ein Telegramm empfangen und im Empfangspuffer abgelegt. Das Telegramm muss vom Anwenderprogramm der SPS durch Setzen des EQ-Bit quittiert werden. Erfolgt dies nicht (SPS wird angehalten) wird das Telegramm über ein TimeOut gelöscht.
3	1	1	Das Anwenderprogramm der SPS hat das Telegramm verarbeitet und quittiert.
4	0	0	Der Empfangspuffer ist wieder verfügbar.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Die **markierten** Werte stehen nicht beim Abschluss eines Sende-/Empfangsvorganges an und dienen der Diagnose zur Laufzeit.

Sendestatus _TXSTATE

Wert	Bedeutung
0	Kein Fehler
20	Initialisierung beim Systemstart. Dieser Status wird einmal im Einschaltmoment gesetzt
21	Sendedatenlänge ungültig Es wurde keine Länge oder eine zu große Länge in TXLEN geladen. Die maximale Länge wird in NetPro in der Projektierung des Sendepuffer festgelegt.
22	Der Sendevorgang ist gescheitert, obwohl im Output-Window der Verbindung für die Sendedaten genügend Platz vorhanden ist.
23	Der Verbindungsaufbau zur Partnerstation ist gescheitert.
24	Die Kombination von SB/SQ ist nicht zulässig. Werden SB/SQ zur Laufzeit in nicht zulässiger Art geändert, kann es zu nicht vorhersehbaren Kommunikationsproblemen führen. Die Ursache für diesen Fehler muss im Anwenderprogramm der SPS abgestellt werden.
25	SB/SQ wurde während eines laufenden Sendevorganges vom Anwenderprogramm der SPS geändert. Werden SB/SQ zur Laufzeit in nicht zulässiger Art geändert, kann es zu nicht vorhersehbaren Kommunikationsproblemen führen. Die Ursache für diesen Fehler muss im Anwenderprogramm der SPS abgestellt werden.
26	Der Sendepuffer kann nicht gelesen werden, da die Struktur zuvor nicht erfolgreich mit dem SFB692 beim PDnetIP-Controller registriert wurde.
27	Die Verbindung zur Partnerstation wurde geschlossen.
28	Beim Start des Sendevorganges durch das Anwenderprogramm der SPS (SB=1, SQ=0) initialisiert der PDnetIP-Controller einen Timer. Kann innerhalb dieser Zeit der Sendevorgang nicht erfolgreich abgeschlossen werden, bricht der PDnetIP-Controller den Sendevorgang ab. Der Timer ist von der Firmware-Version abhängig und beläuft sich auf ca. 10 Sekunden.
29	Statusmeldung: Die Sendedaten liegen bereit und werden an die Partnerstation übertragen.
30	Statusmeldung: Der PDnetIP-Controller hat den Start eines Sendevorganges durch das Anwenderprogramm der SPS erkannt.
31	Statusmeldung: Der PDnetIP-Controller wartet auf die Sendedatenlänge.
32	Statusmeldung: Der Sendevorgang wurde gestartet und der PDnetIP-Controller wartet auf den Sendepuffer.
33	Statusmeldung: Der Sendevorgang wurde gestartet und der PDnetIP-Controller wartet auf die Strukturkennung, um den Sendepuffer zu bestimmen.
34	Die in Wort TXSTRUC übergebene Strukturkennung ist in der Projektierung nicht enthalten.
35	Die Kommunikation zur Partnerstation ist zu langsam. Der PDnetIP-Controller kann die Daten nicht so schnell zustellen wie diese beauftragt werden.
36	Sendedatenlänge zu groß. Der PDnetIP-Controller kann in der Übertragungsart „Header mit Kennung“ maximal 510 Bytes senden.
37	Der Sendevorgang ist gescheitert, da keine Verbindung aufgebaut wurde und keine Partnerstation projektiert ist.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Wert	Bedeutung
38	Optimiere Datenübertragung wurde projektiert aber der Datenpuffer ist noch ungültig.
39	Es ist kein automatischer Verbindungsaufbau projektiert. Da keine Verbindung zu einer Partnerstation bestand wurde der Sendeauftrag negativ quittiert.

Empfangsstatus _RXSTATE

Wert	Bedeutung
0	Kein Fehler
50	Initialisierung beim Systemstart. Dieser Status wird einmal im Einschaltmoment gesetzt
51	Es liegen mehr Empfangsdaten vor als im Empfangspuffer abgelegt werden können. Es werden soviel Daten empfangen wie in den Empfangspuffer der SPS passen. Die verbleibenden Daten werden im nächsten Empfangszyklus im Empfangspuffer übergeben.
52	Die Verbindung zur Partnerstation wurde geschlossen.
53	Der Empfangspuffer kann nicht beschrieben werden, da die Struktur zuvor nicht erfolgreich mit dem SFB692 beim PDnetIP-Controller registriert wurde.
55	Die Kombination von EB/EQ ist nicht zulässig. Werden EB/EQ zur Laufzeit in nicht zulässiger Art geändert, kann es zu nicht vorhersehbaren Kommunikationsproblemen führen. Die Ursache für diesen Fehler muss im Anwenderprogramm der SPS abgestellt werden.
56	Beim Start des Empfangsvorganges durch den PDnetIP-Controller (Daten wurden empfangen) initialisiert der PDnetIP-Controller einen Timer. Kann innerhalb dieser Zeit der Empfangsvorgang nicht erfolgreich abgeschlossen werden, bricht der PDnetIP-Controller den Empfangsvorgang ab. Der Timer ist von der Firmware-Version abhängig und beläuft sich auf ca. 10 Sekunden.
57	Statusmeldung: Der PDnetIP-Controller wartet auf das Durchschreiben der Empfangslänge RXLEN.
58	Statusmeldung: Der PDnetIP-Controller wartet auf das Durchschreiben der Empfangsdaten.
59	Die empfangene Strukturkennung ist in der Projektierung nicht enthalten. Daher ist es dem PDnetIP-Controller nicht möglich, den Daten einen Empfangspuffer zuzuordnen. Die Verbindung wird darauf hin geschlossen.
60	Der Empfangsvorgang für den Header ist gescheitert, obwohl im Input-Window der Verbindung die Daten verfügbar sind.
61	Der Empfangsvorgang wurde abgebrochen, da ein leerer Sendepuffer (Größe=0) konfiguriert wurde.
62	Der Empfangsvorgang ist gescheitert, obwohl im Input-Window der Verbindung Daten vorhanden sind.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Verbindungsstatus (_STATE)

Wert	Bedeutung
1	Es besteht keine Verbindung zu einer Partnerstation.
2	Der PDnetIP-Controller baut eine Verbindung zur Partnerstation auf.
3	Der PDnetIP-Controller besitzt eine Verbindung zur Partnerstation.
9	Verbindungsstatus unbekannt

Die Gesamtgröße der Auftragspuffer darf den zur Verfügung stehenden Arbeitsspeicher des PDnetIP-Controllers nicht überschreiten. Ein Auftragspuffer sollte die Größe von 4.096 Byte nicht überschreiten.

Ein Sendetelegramm wird nur dann übertragen, wenn bei der Partnerstation genügend Platz im Input-Window des Sockets zur Verfügung steht. Der PDnetIP-Controller arbeitet so, um eine Fragmentierung der Sendedaten zu vermeiden. Bedingt dadurch, dass bei einem „freien“ TCP-Protokoll kein Anfang/Ende eines Telegramms von der Empfangsstation erkannt werden kann, nutzt der PDnetIP-Controller das beschriebene Verhalten. Kann das Telegramm nicht übertragen werden, generiert der PDnetIP-Controller den Fehlercode 35 und beendet den Sendevorgang.

Alle gesendeten/empfangenen Daten werden vom PDnetIP-Controller ohne Veränderung dem Sendepuffer entnommen bzw. im Empfangspuffer abgelegt.

Eine Manipulation der Byte-Order erfolgt nicht, alle Daten verbleiben in ihrem Format. Dies bedeutet, dass die Daten zwischen Socket und Signalspeicher 1:1 kopiert werden.

Wird eine Verbindung zu einer Port-Nummer aufgebaut zu der bereits eine Verbindung besteht, wird die vorhandene Verbindung geschlossen. Zu einem Zeitpunkt kann pro Portnummer nur eine aktive Verbindung vorhanden sein.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Übertragungsart „Header mit Kennung“

In der Übertragungsart „Header mit Kennung“ erfolgt keine statische Zuordnung der Sende-/Empfangspuffer. Über die in der NetPro-Projektierung des TCP-Interfaces zu erstellenden Tabelle „Strukturen“ entnimmt die Firmware alle benötigten Informationen, wo die Sende-/Empfangsdaten in der SPS liegen.

Die Tabelle „Strukturen“ beinhaltet für jeden Eintrag die folgenden Informationen:

- **Protokoll-Kennung**. Die Protokoll-Kennung ist die im Header übertragene Kennung (LB/HB), welche den Inhalt der Daten klassifiziert.
- **Struktur-Kennung**. Die Struktur-Kennung identifiziert die der Protokoll-Kennung zugeordneten Sende-/Empfangspuffer im Signalspeicher der Steuerung. Die Strukturen müssen zuvor mit dem SFB692 (PDSTR) gegenüber dem PDnetIP-Controller angemeldet werden.
- **Größe**. Die Größe (in Byte) spezifiziert wie groß der über den Socket übertragene Datenbereich ist. Dadurch können mehre hintereinander gesendete Telegramme zerlegt werden.

Der PDnetIP-Controller legt für jeden Eintrag einen Puffer in Sende- und Empfangsrichtung an. **Wichtig: Zu einem Zeitpunkt kann der gleiche Puffer in Empfangsrichtung nur einmal genutzt werden. Das Verhalten muss von den Partnerstationen gewährleistet werden.**

- **Empfangsvorgang**
  Empfängt der PDnetIP-Controller einen Header, schlägt er die im Header enthaltene Struktur-Kennung in der Tabelle „Strukturen“ nach. Aus der Tabelle leitet sich die Adresslage des Empfangspuffers (Struktur-Kennung) ab. Das Anwenderprogramm der SPS kann dem Wort „RXSTRUC“ entnehmen welche Struktur empfangen wurde.
- **Sendevorgang**
  Beauftragt das Anwenderprogramm der SPS einen Sendevorgang, entnimmt der PDnetIP-Controller dem Wort „TXSTRUC“ welche Struktur gesendet wird.

Die Längeninformation wird ebenfalls der Tabelle entnommen. Die Worte RXLEN und TXLEN haben in dieser Übertragungsart keine Bedeutung.

Achtung: In der Übertragungsart „Header mit Kennung“ dürfen die gesendeten Strukturen maximal 510 Byte groß sein.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Option „Zeitoptimierte Kommunikation zum Endgerät“

In dieser Betriebsart werden pro SPS-Zyklus alle Sende- und Statusdaten aus dem Signalspeicher der Steuerung in den PDnetIP-Controller übertragen. Dies bedingt dass alle Sende- und Statusdaten in einem Zyklus gelesen werden können und nur für ein begrenzt großes Datenmodell zur Verfügung stehen kann. Wie groß das Datenmodell sein darf, hängt von der Projektierung des Endgeräteinterfaces und einer evtl. vorhandenen ICOS-Projektierung ab.

Achtung: Wird die zulässige Größe überschritten arbeitet der Treiber fehlerhaft!

Option „Beauftragung ohne Quittungsbehandlung“

In dieser Betriebsart arbeitet der PDnetIP-Controller ohne Quittungsbit und es gelten andere Regeln für die Verwendung von SB/SQ sowie EB/EQ. Um diese Betriebsart nutzen zu können muss die Option „Zeitoptimierte Kommunikation zum Endgerät“ aktiviert werden.

Am Anfang des SPS Zyklus müssen alle Sendebits gelöscht und am Ende des SPS Zyklus alle Empfangsbits gelöscht werden. Diese Betriebsart ermöglicht es dem PDnetIP-Controller pro SPS-Zyklus ein Telegramm zu senden. Das Anwenderprogramm der SPS bekommt keine Meldung darüber, ob ein Sendevorgang erfolgreich war.

Option „KEEP ALIVE senden“

Wird diese Option aktiviert, sendet der PDnetIP-Controller zyklisch ein KEEP-ALIVE an die Partnerstation und die Zeitüberwachung der Verbindung auf Inaktivität der Kommunikation wird deaktiviert. Eine in dieser Betriebsart geöffnete Verbindung kann somit solange geöffnet bleiben wie die Partnerstation erreichbar ist. Andernfalls baut der PDnetIP-Controller nach einer Zeit eine nicht genutzte Verbindung ab.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

2.2.2.5 MSI-Schnittstelle

Der PDnetIP-Controller unterstützt **optional** das „MSI TCP/IP Interface“ als Server. Der Datenaustausch erfolgt über eine SEND/RECEIVE-Schnittstelle. Dazu benötigt der PDnetIP-Controller mit Hilfe des PDNIP-Bausteins Zugriff auf den Signalspeicher der Steuerung.

Die zur Verarbeitung notwendigen Steuerinformationen werden vorab mit NetPro projektiert, und im nichtflüchtigen Setupdatenspeicher des PDnetIP-Controllers abgelegt.

Die MSI-Projektierungsinformationen werden in NetPro als eine Liste von Verbindungen erstellt. Jede Verbindung besitzt folgende Eigenschaften:

- Eine Kennung (1..65535) zur eindeutigen Identifikation der Verbindung.
- Ein Name (24 Zeichen) zur symbolischen Identifikation der Verbindung.
- Die Portnummer (15000.15999) unter welcher der Server erreichbar ist.
- Die Adresslage des Datenpuffers im Signalspeicher der Steuerung.

Jede projektierte Verbindung belegt im Signalspeicher der Steuerung vier Merker, sowie sieben Worte. Über diese Daten kommuniziert das Anwenderprogramm der SPS mit der MSI-Schnittstelle der Firmware.

Zur Ablage dieser Informationen ist vorab eine Startadresse im Wort- sowie eine Startadresse im Merkerbereich der Steuerung zu definieren. Die Adressen werden von der MSI-Schnittstelle automatisch belegt.

NetPro exportiert die Belegung der Signalspeicheradressen als ASD-Datei. Die so exportierten Informationen können direkt mit AKF oder ALD eingelesen werden.

Jeder in eine ASD-Datei exportierte Symbolname setzt sich aus dem Verbindungsnamen, sowie einem Suffix zusammen. Im Anwenderprogramm der SPS kann über die symbolischen Namen einfach auf die Informationen zugegriffen werden.

Nach jeder Veränderung der MSI-Projektierung (Verbindung einfügen oder löschen), kann sich die Belegung des Signalspeichers verändern.

Die folgende Tabelle beschreibt die Bedeutung der für jede Verbindung belegten Status-Worte und Merker. Der Suffix ergänzt den mit NetPro projektierten Verbindungsnamen.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Suffix	Gruppe	Zustand
'_ID'	Global	Die mit NetPro projektierte Verbindungskennung.
,_CSTATE"		Der aktuelle Verbindungsstatus.
'_ERROR'		Fehlercode warum die Verbindung geschlossen wurde
'_SB'	Sende-richtung	Beauftragungsbit in Senderichtung. Über das Sendebit zeigt das Anwenderprogramm der SPS, dass ein Sendetelegramm vorliegt.
'_SQ'		
'_TXSTATE'		Letzter/Aktueller Status in Senderichtung.
,_TXLEN'		Anzahl zu sendender Bytes
'_EB'	Empfangs-richtung	Das Empfangsbit informiert das Anwenderprogramm der SPS über ein neues Empfangstelegramm.
'_EQ'		
'_RXSTATE'		Letzter/Aktueller Status in Empfangsrichtung.
,_RXLEN'		Anzahl empfangener Bytes

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Die nachfolgende Legende zeigt die im folgenden genutzte farbliche Kodierung der Seite, die für das Setzen des Zustandes verantwortlich ist.

Der Zustand wird vom PDnetIP-Controller gesetzt und dient dem Anwenderprogramm der SPS als Triggerbedingung.

Der Zustand wird vom Anwenderprogramm der SPS gesetzt und dient der Firmware des PDnetIP-Controllers als Triggerbedingung.

Der Zustand ist nicht zulässig und wird von einem fehlerhaften Anwenderprogramm der SPS verursacht.

Mögliche Zustände SB/SQ

SB	SQ	Zustand
0	1	Sendepuffer frei. Dieser Zustand wird vom PDnetIP-Controller auch im Einschaltmoment gesetzt, so dass im Anwenderprogramm der SPS keine Normierung der SB/SQ-Merker durchgeführt werden muss.
1	0	Wird vom Anwenderprogramm der SPS gesetzt, wenn ein Telegramm gesendet werden soll. Wurde der Sendevorgang angestoßen, darf dieser vom Anwenderprogramm der SPS nicht wieder gelöscht werden. In jedem Fall muss das Anwenderprogramm der SPS auf die Reaktion des PDnetIP-Controllers warten.
0	0	Zustand ist nicht zugelassen und wird vom PDnetIP-Controller auf SB=0 und SQ=1 normiert.
1	1	

Ablauf Sendevorgang

Schritt	SB	SQ	Zustand
1	0	1	Der Sendepuffer ist verfügbar. Der Zustand wird von dem PDnetIP-Controller initialisiert. Auf diesen Zustand muss das Anwenderprogramm der SPS warten, um einen Sendevorgang starten zu können.
2	1	0	Das Anwenderprogramm der SPS hat den Sendepuffer mit Daten gefüllt und startet so den Sendevorgang.
3	0	1	Der PDnetIP-Controller hat den Sendewunsch bearbeitet und der Sendepuffer steht wieder zur Verfügung. Ob der Sendevorgang erfolgreich war, kann TXSTATE entnommen werden.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Mögliche Zustände EB/EQ

EB	EQ	Zustand
0	0	Empfangspuffer frei. Dieser Zustand wird vom PDnetIP-Controller im Einschaltmoment gesetzt, so dass im Anwenderprogramm der SPS keine Normierung der EB/EQ-Merker durchgeführt werden muss.
0	1	Zustand ist nicht zugelassen und wird vom PDnetIP-Controller normiert.
1	0	Der PDnetIP-Controller hat ein Telegramm empfangen und im Empfangspuffer abgelegt. Das Telegramm muss vom Anwenderprogramm der SPS durch Setzen des EQ-Bit quittiert werden. Erfolgt dies nicht (SPS wird angehalten), wird das Telegramm über ein TimeOut gelöscht.
1	1	Das Anwenderprogramm der SPS hat das Telegramm verarbeitet und quittiert.

Ablauf Empfangsvorgang

Schritt	EB	EQ	Zustand
1	0	0	Empfangspuffer frei. Dieser Zustand wird vom PDnetIP-Controller im Einschaltmoment gesetzt, so dass im Anwenderprogramm der SPS keine Normierung der EB/EQ-Merker durchgeführt werden muss.
2	1	0	Der PDnetIP-Controller hat ein Telegramm empfangen und im Empfangspuffer abgelegt. Das Telegramm muss vom Anwenderprogramm der SPS durch Setzen des EQ-Bit quittiert werden. Erfolgt dies nicht (SPS wird angehalten) wird das Telegramm über ein TimeOut gelöscht.
3	1	1	Das Anwenderprogramm der SPS hat das Telegramm verarbeitet und quittiert.
4	0	0	Der Empfangspuffer ist wieder verfügbar.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Verbindungsstatus (_CSTATE)

Wert	Bedeutung
1	Es besteht keine Verbindung zu einer Partnerstation.
2	Der PDnetIP-Controller besitzt eine Verbindung zur Partnerstation.

Fehlercode (_ERROR)

Wert	Bedeutung
0	Ok
1	Die bestehende Verbindung wurde geschlossen da eine neue Verbindung aufgebaut wird.
2	Die Verbindung wurde aufgrund eines Kommunikationsfehler geschlossen (Exception-List).
3	Die Verbindung ist abgerissen und wurde geschlossen.
4	Die verbindung wurde von der Partnerstation beendet.
5	Eine empfangene Nachricht beinhaltet im Header ungültige Zeichen und entspricht nicht der Protokollspezifikation.
6	Die Verbindung wurde aufgrund eines Kommunikationsfehlers in Senderichtung geschlossen.
7	Die Verbindung wurde geschlossen da bei einem Sendetelegramm die Anzahl zulässige Wiederholungen erreicht ist und keine positive Quittung der Partnerstation eingetroffen ist.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Sendestatus _TXSTATE

Wert	Bedeutung
0	Ok
31	Die Datenstruktur des Sendepuffers wurde nicht angemeldet.
32	Die Sendelänge ist ungültig. Die Länge muss größer Null sein und darf nicht größer als der Sendepuffer sein.
33	Timeout; Sendedaten wurden nicht gültig
34	Timeout; Sendedatenlänge wurde nicht gültig
35	Timeout; Die Sendedaten konnten nicht rechtzeitig gesendet werden
36	Die Anzahl Sendewiederholungen ist abgelaufen ohne eine positive Quittung der Partnerstation zu empfangen
37	Error Code 03 von Partnerstation empfangen
38	Ein im MSI-Protokoll nicht definierter Fehlercode wurde von Partnerstation empfangen.
39	Es wurde ein Sendeauftrag ohne eine bestehende Verbindung gestartet.
40	Kollision: Es wurde ein Sendeauftrag beauftragt und zeitgleich liegt ein Empfangstelegramm vor. Der Sendeversuch muss zu einem späteren Zeitpunkt wiederholt werden.
41	Der Datenbereich enthält nicht zulässige Zeichen.

Empfangsstatus _RXSTATE

Wert	Bedeutung
0	Ok
61	TimeOut; Empfangsdaten nicht in SPS geschrieben
62	TimeOut; Die Empfangsdaten wurde vom Anwenderprogramm der SPS nicht quittiert (EQ wurde nicht gesetzt)
63	TimeOut; Quittung konnte nicht rechtzeitig gesendet werden
64	Die Datenstruktur des Empfangspuffers wurde nicht angemeldet.
65	Das Empfangstelegramm ist größer als der Empfangspuffer.
66	Daten wurden bei stehender SPS empfangen.
67	Der Datenbereich enthält nicht zulässige Zeichen.

Alle gesendeten/empfangenen Daten werden vom PDnetIP-Controller ohne Veränderung dem Sendepuffer entnommen bzw. im Empfangspuffer abgelegt.

Eine Manipulation der Byte-Order erfolgt nicht, alle Daten verbleiben in ihrem Format.

Wird eine Verbindung zu einer Port-Nummer aufgebaut zu der bereits eine Verbindung besteht, wird die vorhandene Verbindung geschlossen. Zu einem Zeitpunkt kann pro Portnummer nur eine Verbindung vorhanden sein.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

SYSLOG-Meldungen

Wurde der SYSLOG-Dienst aktiviert, so erzeugt der MSI-Treiber im Bedarfsfall für eine vereinfachte Diagnose Statusmeldungen. Jede Syslog-Meldung welche sich auf eine Verbindung bezieht beinhaltet am Anfang den Namen und die projektierte ID der Kennung.

'MSI: No license'
Die Größe des Sende- oder Empfangspuffers kann nicht auf eine ganze Anzahl Elemente des gewählten Datentyps abgebildet werden.

'size-error'
Die Größe des Sende- oder Empfangspuffers kann nicht auf eine ganze Anzahl Elemente (Sende-/Empfangspuffer) des projektierten Datentyps abgebildet werden.

'value %s fixed'
Ein Status-Wort oder Status-Merker wurde vom Anwenderprogramm der SPS unerwartet verändert und wurde vom PdnetIP-Controller korrigiert.

'protocol error, user data received while last processing not completed'
Es wurden ein erneutes Telegramm mit Anwenderdaten empfangen ohne das für das vorige Telegramm die Transaktion beendet werden konnte.

'protocol error, duplicated flow control'
Es wurde ein Telegramm mit Anwenderdaten empfangen welches eine zuvor bereits genutzten Wert in „Flow Control“ wiederholt benutzt.

'protocol error, data contain invalid data'
Das empfangene Telegramm beinhaltet ungültige Daten.

'user data received with stopped plc'
Die Steuerung steht und kann die Empfangsdaten nicht verarbeiten.

'receive buffer address unknown'
Die Datenstruktur des Empfangspuffers wurde nicht angemeldet.

'receive data bigger than buffer size'
Das Empfangstelegramm ist größer als der Empfangspuffer.

'protocol error, ack received without previous frame'
Es wurde ein Quittung empfangen ohne das zuvor ein Telegramm gesendet wurde welches noch quittiert werden müsste.

'protocol error, invalid len (ack)'
Die empfange Quittung hat eine nicht zulässige Länge.

'protocol error, undefinied errror code'
Der im empfangenen Quittungstelegramm übermittelte “Error Code” ist nicht definiert und unzulässig.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

'protocol error, invalid ka'
Das empfangen Lebenszeichen (KA) hat eine nicht zulässige Länge.

'tx-length invalid'
Die Sendelänge ist ungültig. Die Länge muss größer Null sein und darf nicht größer als der Sendepuffer sein.

'send buffer address unknown'
Die Datenstruktur des Sendepuffers wurde nicht angemeldet.

'protocol error, not allowed response received (ack for keep alive)'
Die empfangene Quittung für ein Lebenszeichen ist nicht zulässig da diese nicht angefordert wurde.

'protocol error, undefined datagram type received'
Es wurde ein Telegramm mit einem unzulässigen Type-Information empfangen.

'receive-error %d'
Es kam zu einem Kommunikationsfehler während der Datenübertragung.

'receiving aborted'
Der Empfangsvorgang wurde abgebrochen da dieser nicht innerhalb der zulässigen Zeit abgeschlossen werden konnte.

'transmit aborted'
Der Sendevorgang wurde abgebrochen da dieser nicht innerhalb der zulässigen Zeit abgeschlossen werden konnte.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

2.2.2.6 Kundenspezifische Treiber

Es sind in der Firmware des PDnetIP-Controllers weitere kundenspezifische Treiber implementiert. Die entsprechende Dokumentation erfolgt dann in den Kunden- bzw. projektspezifischen Dokumenten.

Benötigen Sie ein spezielles Protokoll oder eine Erweiterung einer vorhandenen Implementierung dann sprechen Sie uns darauf an.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

2.2.3 Serielle Protokolle

Der PDnetIP-Controller unterstützt eine Reihe serieller Protokolle zur Kommunikation mit anderen Geräten über offene Schnittstellen sowie kundenspezifische Implementierungen als Sonderlösung.

2.2.3.1 Access-Server

Der Access-Server ermöglicht die Kommunikation mit einer seriellen Schnittstelle über einen wählbaren TCP-Port zu nutzen. Die Partnerstation kann dazu beispielsweise einen virtuellen ComPort-Treiber nutzen.

2.2.3.2 IS-Tester

Das IS-Tester-Protokoll basiert auf der Schnittstellenspezifikation in Version 037.1 vom 27.04.2005 und implementiert ein serielles Protokoll zur Kommunikation zwischen einem RPS und einem IS-Tester.

Der PDnetIP-Controller wickelt dabei die serielle Kommunikation ab und überträgt zwischen RPS und IS-Tester vollständige Telegramme in Sende- und Empfangsrichtung. Die Prüfsumme wird automatisch gebildet und geprüft. Das Anwenderprogramm der SPS kommuniziert über Sende- und Empfangsbit sowie Sende- und Empfangspuffer mit dem Treiber.

Der Treiber wurde nicht für einen Mischbetrieb geplant oder getestet!

Das IS-Tester-Protokoll darf zu einem Zeitpunkt nur auf einer seriellen Schnittstelle des PDnetIP-Controllers genutzt werden.

Werden zugleich weitere Treiber (incl. VDM) genutzt, muss gewährleistet sein, dass immer alle Daten in einem SPS-Zyklus gekoppelt werden können.

Die serielle Kommunikation erfolgt mit 38.400 Baud (8N1). Das Protokoll sieht kein Hardware-Handshake vor, was dazu führen kann, dass Datenverlust auftritt. In diesen Fällen kann zwangsläufig das geforderte Timing nicht eingehalten werden.

Die in der Spezifikation definierte Request/Response-Ablauf zwischen RPS und IS-Tester muss durch das Anwenderprogramm der SPS gewährleistet werden. Der PDnetIP-Controller prüft den Datenstrom nicht auf derartige Regelverletzungen. Jedes Setzen des Sendebit löst einen Sendevorgang aus.

Die Implementierung unterstützt die Quellkennungen 0x40 für den RPS sowie 0x80 für den IS-Tester. Ferner werden die Telegrammtypen 0xAA, 0x01 sowie 0xFA unterstützt.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

In einem Testaufbau konnte bei einer SPS-Zykluszeit von 16ms pro SPS-Zyklus erfolgreich ein Telegramm gesendet werden.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Mit NetPro werden die folgenden Parameter projektiert:

- **Sendepuffer**
  Im Sendepuffer werden vom Anwenderprogramm der SPS die zu sendenden Daten abgelegt. Die Daten liegen immer im Wort-Bereich der Steuerung. Der Sendepuffer belegt ab der projektierten Adresse 59 Worte.
- **Sendebit**
  Das Sendebit informiert den PDnetIP-Controller darüber, dass im Sendepuffer ein Telegramm liegt, welches gesendet werden soll.
- **Empfangspuffer**
  Im Empfangspuffer werden vom PDnetIP-Controller alle empfangenen Daten abgelegt. Die Daten liegen immer im Wort-Bereich der Steuerung. Der Empfangspuffer belegt ab der projektierten Adresse 59 Worte.
- **Empfangsbit**
  Wurde vom PDnetIP-Controller ein Telegramm empfangen, so wird das Anwenderprogramm der SPS durch Setzen des Empfangsbit darüber informiert.
- **Aktiv-Bit**
  Das Aktiv-Bit zeigt dem Anwenderprogramm der SPS an, ob eine aktive Kommunikation zum IS-Tester vorhanden ist. Die Information wird aus den Empfangsdaten generiert und nach Inaktivität der Kommunikation (TimeOut) automatisch gelöscht.
- **Timeout-Zeit**
  Jedes Sendetelegramm der SPS an den IS-Tester lädt das Telegramm-Timeout. Durch die in der Spezifikation festgelegte Request/Response-Regel kann das Telegramm-Timeout geprüft werden, da nach einem Empfangstelegramm immer ein Telegramm gesendet wird. Wird diese Regel nicht berücksichtigt, kann das Telegramm-Timeout nicht richtig gebildet werden.
- **Gesamt-TimeOut-Zeit**
  Jedes Empfangstelegramm lädt das Gesamt-Timeout.
- **Diagnose-Spur**
  In der Diagnose-Spur stellt der PDnetIP-Controller dem Anwenderprogramm der SPS eine Reihe von Informationen zur Fehlersuche bereit. Die Diagnose-Spur belegt ab der projektierten Adresse 10 Worte.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Tritt während des Betriebs ein TimeOut auf (TimeOut oder Gesamt-Timeout) beginnt die LED der seriellen Schnittstelle zu blinken. Durch drücken des Tasters (ca. 1 Sekunde) wird der Fehler quittiert und das Blinken hört auf, wenn der Fehler nicht mehr ansteht.

Die Diagnose-Spur ist wie folgt aufgebaut:

Wort	Bedeutung
1	Anzahl vom Anwenderprogramm der SPS gesendete LogIn-Telegramme
2	Anzahl aufgetretener TimeOut-Ereignisse
3	Anzahl aufgetretener Gesamt-TimeOut-Ereignisse
4	Anzahl Empfangstelegramme
5	Anzahl Sendetelegramme
6	Anzahl gelöschter Empfangszeichen (in Byte) wegen gescheiterter Telegrammprüfung (Prüfsumme)
7	Anzahl vom Anwenderprogramm der SPS mit ungültiger Quellkennung gesendeter Telegramme
8	Anzahl vom Anwenderprogramm der SPS gesendete Telegramme für welche die Telegrammlänge nicht bestimmt werden konnte.
9	Anzahl Unterbrechungen wegen SPS Stop/Start
10	Anzahl gelöschter Empfangsdaten (in Byte) wegen stehender SPS

Wird der rote Taster gedrückt (ca. 1 Sekunde), setzt der PDnetIP-Controller alle Elemente der Diagnose-Spur auf Null.

Zur Gewährleistung der benötigten Reaktionszeit gibt es besondere Anforderungen an die Gestaltung des SPS Programms.

Zyklus-Anfang
SFB 693 (PDNIP) aufrufen Der PDnetIP-Controller tauscht mit dem Baustein-Aufruf alle Daten zwischen dem Signalspeicher der SPS und dem PDnetIP-Controller aus
Sendebit löschen
Wenn das Empfangsbit gesetzt wurde, steht im Empfangspuffer ein Telegramm zur Verfügung.
Zum Senden das Sendebit setzen
Empfangsbit löschen
Zyklus-Ende

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

2.2.3.3 ModBus Master

Wird für eine serielle Schnittstelle ein ModBus Master projektiert wird der Treiber über die ModBus Master Auftragsblöcke gesteuert. Die Projektierung erfolgt innerhalb des ModBus/TCP Dienst im PDnetIP-Kanal. Als IP-Adresse ist 0.0.0.1 für die serielle Schnittstelle 1 und 0.0.0.2 für die serielle Schnittstelle 2 zu nutzen.

Die Transaktions-ID wird vom seriellen ModBus nicht unterstützt.

Im Feld UnitID wird die ModBus-Stationsadresse eingetragen.

Empfängt der ModBus/TCP-Server ein Telegramm und der verwendete UnitIdentifier entspricht der Nummer einer seriellen Schnittstelle mit projektierten ModBus Master und aktivierten Routing, so reicht der PDnetIP-Controller das Telegramm über die serielle Schnittstelle weiter. Eine über die serielle Schnittstelle eintreffenden Antwort wird analog via ModBus/TCP an die Partnerstation gesendet.

Werden mehrere Stationen an einen Master angeschlossen (Party Line) ist es wichtig dass diese keine Pausenzeiten zwischen Senden- und Empfangen erfordern.

2.2.3.4 ModBus Slave

Wird für eine serielle Schnittstelle ein ModBus Slave projektiert stellt dieser alle über das Mapping definierten Bereiche zur Verfügung. Die im Mapping definierten Datenbereiche stehen gleichermaßen über ModBus/TCP und seriellen ModBus Slave zur Verfügung.

Werden mehrere Stationen an einen Master angeschlossen (Party Line) ist es wichtig das diese keine Pausenzeiten zwischen Senden- und Empfangen erfordern.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

2.2.3.5 Seab 1/F Master

Der PDnetIP-Controller unterstützt **optional** das „Seab 1/F Master“ Protokoll. Dazu benötigt er mit Hilfe des PDNIP-Bausteins Zugriff auf den Signalspeicher der Steuerung.

Das Protokoll „Seab 1/F Master“ ermöglicht die Kommunikation zu folgenden Stationstypen und dazu Kompatiblen:

- Micro
- U120
- U250

Eine genaue Beschreibung erfolgt im „Seab 1/F Benutzerhandbuch für PDnetIP-Controller“. Im folgenden werden die Projektierungseinstellung sowie Leistungsparameter der Implementierung kurz dargestellt.

Mit der **lizenzpflichtigen** Vollversion von NetPro werden die Parameter des Treibers und das Datenmodell projektiert.

- **TV, Vorlaufzeit.** Zeitspanne, die nach der Aktivierung von RTS abläuft, bevor das erste Zeichen des Telegramms gesendet wird.
- **TN, Nachlaufzeit.** Zeitspanne, die nach dem Senden des letzten Zeichens abläuft, bevor RTS deaktiviert wird.
- **TP, Pausenzeit.** Zeitspanne nach Deaktivieren von RTS, in der die Stationen und deren Modems die Umschaltung zwischen Senden/Empfangen durchführen.
- **TWM, Wartezeit Master.** Antwortet die Unterstation nicht innerhalb von TWM, wird damit dem Master ein Fehler angezeigt. TWM muss größer als TP+TV sein, damit eine Unterstation ein gültiges Telegramm übertragen kann. Um eine zuverlässige Zustandserkennung zu ermöglichen, müssen TP, TV und TN bei Master und Unterstation identisch sein.
- **TVM, Verzögerungszeit Master.** Wurde vom Master ein fehlerhaftes Telegramm empfangen, so verzögert der Master den nächsten Sendevorgang um TVM, um der Unterstation den Fehler anzuzeigen.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

- **Wiederholung im Fehlerfall.** Wird ein Pollaufruf nicht oder fehlerhaft beantwortet, so wird der Pollaufruf so oft wiederholt, bis eine gültige Antwort eintrifft. Mit der Wiederholungszahl wird dieser Vorgang auf eine maximale Anzahl Pollaufrufe beschränkt. Sinnvolle Werte liegen zwischen 0-3.

Für Unterstationen sind die A-Bytes 1 bis 126 und 127 gültig, wobei die virtuelle Unterstation 127 allein dem Senden von Broadcast an alle Unterstationen dient.

In die Poll-Liste können maximal 126 Unterstationen eingetragen werden, die vom Seab1F-Treiber in dieser Reihenfolge gepollt werden. Die Poll-Liste ist unsortiert, Unterstationen können mehrfach vorhanden sein.

Micro

Objekt	Anzahl / Datentyp
Meldungen	128 / Bit
Echtzeit Meldungen	keine
Echtzeit Meldungen aus dem Ringpuffer	keine
Echtzeit Meldungen aus dem Datenmodell	keine
Zählwerte	24 / Word
Umgespeicherte Zählwerte	24 / Word
Messwerte 8Bit	24 / Byte
Messwerte 16Bit	24 / Word
Verwaltungsmeldungen	48 / Bit
Stationsmeldungen	9 / Byte (APEX Datentyp)
Befehle	64 / Bit
Sollwerte	32 / Int. / Word (digital / analog)
Aufrufe	4 / Byte (APEX Datentyp)

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

U120

Objekt	Anzahl / Datentyp
Meldungen	256 / Bit
Echtzeit Meldungen	256 / Bit
Echtzeit Meldungen aus dem Ringpuffer	256 / Bit
Echtzeit Meldungen aus dem Datenmodell	256/ Bit
Zählwerte	63 / Word
Umgespeicherte Zählwerte	63 / Word
Messwerte 8Bit	64 / Byte
Messwerte 11/16Bit	63 / (11 Bit + Vz) Word
Verwaltungsmeldungen	48 / Bit
Stationsmeldungen	9 / Byte (APEX Datentyp)
Befehle	256 / Bit
Sollwerte	16/32 / Int./Word (digital/analog)
Aufrufe	4 / Byte (APEX Datentyp)

U250

Objekt	Anzahl / Datentyp
Meldungen	1024 / Bit (in Summe mit E. Meldungen)
Echtzeit Meldungen	1024 / Bit (in Summe mit Meldungen)
Echtzeit Meldungen aus dem Ringpuffer	1024 / Bit
Echtzeit Meldungen aus dem Datenmodell	1024 / Bit
Zählwerte	128 / Word
Umgespeicherte Zählwerte	128 / Word
Messwerte 8Bit	256 / Byte (in Summe mit 16Bit Mw.)
Messwerte 16Bit	256 / Byte (in Summe mit 8Bit Mw.)
Verwaltungsmeldungen	48 / Bit
Stationsmeldungen	9 / Byte (APEX Datentyp)
Befehle	512 / Bit Feld
Sollwerte	256 / Int. / Word (digital / analog)
Aufrufe	4 / Byte (APEX Datentyp)

Broadcast

Objekt	Anzahl / Datentyp
Aufrufe	4 / Byte (APEX Datentyp)

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

2.2.3.6 Kundenspezifische Treiber

Es sind in der Firmware des PDnetIP-Controllers weitere kundenspezifische Treiber implementiert. Die entsprechende Dokumentation erfolgt dann in den Kunden- bzw. projektspezifischen Dokumenten.

Benötigen Sie ein spezielles Protokoll oder eine Erweiterung einer vorhandenen Implementierung dann sprechen Sie uns darauf an.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

3 Installation des PAD-250e

3.1 Einbau des PAD-250e

Der PDnetIP-Controller PAD-250e wird in einen Steckplatz eines A250-Magazines eingesetzt. Es können nur Steckplätze im Grundmagazin verwendet werden. Beim Einstecken in das Magazin muss die Baugruppe richtig einrasten. (**Keine Gewalt anwenden!)** Nach Einrasten der Baugruppe müssen die Sicherungsschrauben angezogen werden.

3.2 Sonder-Funktions-Bausteine (SFB)

Um den PDnetIP-Controller vom Anwenderprogramm der SPS ansprechen zu können, sind die dafür notwendigen Sonderfunktionsbausteine im Lieferumfang enthalten. Über die Bausteine werden Daten zwischen dem Signalspeicher der SPS und dem PDnetIP-Controller ausgetauscht. Damit die Bausteine in einem Anwenderprogramm der SPS zur Verfügung stehen, müssen zwei Bausteinpakete zunächst in die Bibliothek von AKF/ALD eingebunden werden. Dieser Vorgang muss einmalig für jede AKF/ALD Installation durchgeführt werden. Nach jeder Installation einer neuen AKF/ALD Version, müssen die Bausteinpakete erneut in die Bibliothek integriert werden.

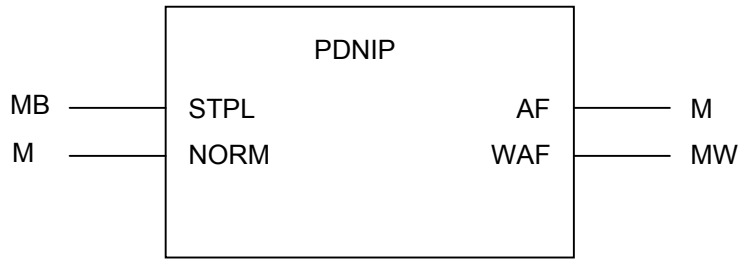
Die Sonderfunktionsbausteine werden für die ICOS-Dienste nicht benötigt.

 © 1992-2014 by APEX® automation technologies GmbH, Braunschweig

3.2.1 SFB693 - PDNIP

Der SFB 693 stellt die Verbindung für alle Datenübertragungen zwischen dem Signalspeicher der ALU und dem PDnetIP-Controller her. Dieser Baustein erhält die zum Datentransfer notwendigen Informationen vollständig vom PDnetIP-Controller, so dass die Parametrierung des Bausteines in verschiedenen Anlagen gleich sein kann. Im Folgenden wird die Bedeutung der Bausteinparameter beschrieben.

SFB 693



Achtung: Der Baustein liegt als NEAR-Version (IM0-Datei) vor. Damit können die übergebenen Parameter STPL, NORM, AF und WAF nicht Element einer Struktur sein.

Achtung: Der Baustein sollte nicht häufiger als einmal pro 10ms aufgerufen werden.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

3 Installation des PAD-250e
3.2 Sonder-Funktions-Bausteine (SFB)

Signal	Bedeutung
STPL	Steckplatz In dem hier eingetragenen Merkerbyte wird der Steckplatz des PDnetIP-Controllers übergeben. (sinnvolle Werte 2-9). Wird als Parameter 253 übergeben erscheint in WAF die Baustein-Version des SFB693.
NORM	Normierung Über diesen Merker informiert das Anwenderprogramm der SPS den Baustein über einen Neu- oder Weiterstart der Steuerung. Im PDnetIP ist der NORM-Eingang nicht von Bedeutung und zur Kompatibilität mit ältern Anlagen vorhanden.
AF	Ausgang Fehler Dieser Merker wird dann vom SFB gesetzt, wenn im Laufe der gerade vorangegangenen Bearbeitung ein Fehler aufgetreten ist. Wenn beim nächsten Aufruf kein Fehler festgestellt wird, wird dieser Merker zurückgesetzt. (Ausgang nicht speichernd)
WAF	Wort Ausgang Fehler In diesem Merkerwort legt der SFB einen Fehler-Code ab, der den aufgetretenen Fehler näher beschreibt. Dieses Merkerwort wird vom SFB selbst nie zurückgesetzt (Ausgang speichernd).

Parameter STPL	Bedeutung des Parameters
2..9	Baustein greift auf den Steckplatz zu.
254	Interne Sonderfunktion im Baustein.
255	Baustein-Version wird an WAF ausgegeben. Die Version wird im Format x.y verwaltet. Im Hi-Byte steht der Versionsanteil X und im Low-Byte der Versionsanteil Y.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Fehlercode	Bedeutung
1	Der übergebene Steckplatz ist ungültig.
7	Der Baustein hat einen Auftrag mit einem ihm nicht bekannten Datentyp erhalten.
8	Auftragsliste defekt
9	Steckplatz nicht aufschaltbar (PAB-Error)
51	PAD-250e im angegebenen Steckplatz nicht gefunden. Der PDnetIP-Controller kann von dem Baustein nur gefunden werden, wenn dieser mit einem entsprechenden Interface projektierte wurde. Dieser Fehlercode tritt nur in Bausteinen auf, welche das Paging nicht unterstützen.
52	Interner Fehler, das dynamische Offset liegt nicht auf Segment-Grenze.
53	Interner Fehler, das dynamische Offset ist noch nicht bekannt, da die ICOS-Kommunikation noch nicht initialisiert wurde.
54	Es wurde ein ICOS-Interface projektiert, aber noch nicht initialisiert. Der Baustein kann erst mit einem initialisierten ICOS-Interface arbeiten, da andernfalls der Speicherbereich für den Datenaustausch nicht bestimmt werden kann.
55	Durch die gewählte Projektierung des ICOS-Interface verbleibt nicht genügend Speicher für den Datenaustausch zwischen PDnetIP-Controller und Steuerung.
56	Es wurde in der Projektierung des Endgerätetreiber im NetPro-Projekt der SFB691 gewählt, jedoch in der Steuerung der SFB693 (PDNIP) aufgerufen. Damit kann von dem Baustein kein geeignetes Interface gefunden werden.
57	Dieser Fehler erscheint, wenn nicht der SFB693 sondern der SFB691 aufgerufen wurde.
58	Eine laufende Firmware wurde vom Anwender (Taster) beendet und der Lader aktiviert.
59	PAD-250e im angegebenen Steckplatz nicht gefunden. Der PDnetIP-Controller kann von dem Baustein nur gefunden werden, wenn dieser mit einem entsprechenden Interface projektierte wurde. Dieser Fehlercode tritt nur in Bausteinen auf, welche das Paging unterstützen.
60	Beim Paging ist ein Fehler aufgetreten.
61	Abbruch der Verarbeitung. Dieser Fehler tritt bei einem Weiterstart der ALU oder nach einem Laden des PAD-250e auf. Alle Strukturen müssen neu angemeldet werden.
62	ALU wurde neu gestartet, alle Strukturen müssen neu angemeldet werden.
30001	Der PDnetIP-Controller hat einen Neustart durchgeführt, alle Strukturen müssen neu angemeldet werden.
30002	Die Verbindung zwischen PDnetIP-Controller und SPS war unterbrochen (Abbruch durch TimeOut), alle Strukturen müssen neu angemeldet werden.
30003..30255	Diese Fehlercodes sind von der eingesetzten Firmware abhängig und werden von der Firmware verwaltet.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

3.2.2 SFB692 - PDSTR

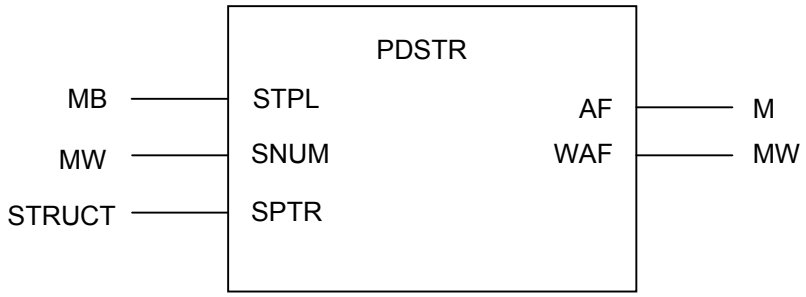
Der SFB 692 wird benötigt, wenn mit dem PDnetIP-Controller Strukturen übertragen werden. Der Baustein macht die Adresse einer Struktur im Signalspeicher der ALU beim PAD-250e bekannt.

Nach einem Neustart der ALU müssen alle in der Kommunikation genutzten Datenstrukturen registriert werden. Der erste Aufruf muss mit der ID Null erfolgen, um eine Initialisierung der Adresstabelle im PAD-250e zu initiieren.

Eine effektive Variante den Aufruf des SFB692 zu steuern liegt darin, sowohl den Fehlerausgang AF des SFB693 auf eine fallende Flanke als auch den NORM-Merker der SPS auf eine fallende Flanke zu überwachen.

Im Einschaltmoment werden sowohl in der ALU als auch im PAD-250e Initialisierungen durchgeführt. Wann diese abgeschlossen sind, lässt sich im Vorfeld nicht sicher sagen. Daher ist es am einfachsten, den Fehlerausgang AF des SFB693 auf eine fallende Flanke zu überwachen und mit einer Zeitverzögerung die Anmeldung aller Datenstrukturen durchzuführen. Durch diese Aufrufvariante wird sowohl der Einschaltmoment als auch ein Neustart des PDnetIP-Controllers berücksichtigt. Werden die Setupdaten des PAD-250e bei laufender SPS neu geladen, so werden auch die Strukturen automatisch angemeldet.

SFB 692



Achtung: Der Baustein liegt ausschließlich als FAR-Version (IM2-Datei) vor. Damit können ausschließlich Strukturen adressiert werden, welche als FAR definiert wurden.

Achtung: Ab Firmware 4.10.16.18 und Baustein 1.12 beträgt die maximale Laufzeit des Bausteines 20ms.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Signal	Bedeutung
STPL	Steckplatz In dem hier eingetragenen Merkerbyte wird der Steckplatz des PDnetIP-Controllers übergeben. (sinnvolle Werte 2-9)
SNUM	Struktur-Kennung In dem Parameter SNUM wird die Kennung der in der Projektierung von NetPro definierten Struktur übergeben. Die Struktur-Kennung Null löscht alle bisher im PDnetIP-Controller gespeicherten Adressinformationen. Der PDnetIP-Controller arbeitet unabhängig von der ALU der A250. Immer dann, wenn sich die Adresslage von Strukturen in der A250 geändert hat oder die ALU neu gestartet wurde, müssen alle Strukturen bei dem PAD-250e registriert werden.
SPTR	Referenz der Struktur In diesen Parameter wird die Struktur (und damit die Adresse der Struktur) an den PAD-250e übergeben.
AF	Ausgang Fehler Dieser Merker wird dann vom SFB gesetzt, wenn im Laufe der gerade vorangegangenen Bearbeitung ein Fehler aufgetreten ist. Wenn beim nächsten Aufruf kein Fehler festgestellt wird, wird dieser Merker zurückgesetzt. (Ausgang nicht speichernd)
WAF	Wort Ausgang Fehler In diesem Merkerwort legt der SFB einen Fehler-Code ab, der den aufgetretenen Fehler näher beschreibt. Dieses Merkerwort wird vom SFB nicht zurückgesetzt (Ausgang speichernd).

Fehlercode	Bedeutung
1	Parameter Steckplatz außerhalb der erlaubten Grenzen
2	Kein PAD-250e mit Strukturinterface im angegebenen Steckplatz gefunden.
110	Die Firmware des PAD-250e unterstützt keine Kommunikation von Datenstrukturen.
111	Ablauffehler der Beauftragung (alte Beauftragung vorhanden)
112	Ablauffehler der Beauftragung (Semaphore ist ungültig)
113	Ablauffehler der Beauftragung (verzögerte Antwort vorhanden)
120	Die Strukturanmeldung wurde vom PAD-250e innerhalb der Wartezeit nicht erfolgreich quittiert.
130	Der Echtzeitkommunikationspuffer für Strukturen ist voll bzw. momentan überlastet.

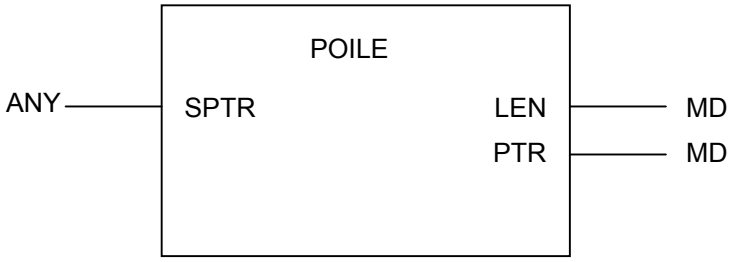
© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

3.2.3 SFB695 – POILE

Der Baustein SFB695 ist Bestandteil des PMCK-Bausteinpaketes und gehört nicht zum Lieferumfang des PDnetIP-Controllers. Das PMCK-Bausteinpaket wurde als Erweiterung zum PDnetIP-Controller PAD-250e zur Kommunikation mit PMCK-Systemen (SICALIS) über das Protokoll RFC1006 entwickelt.

Der SFB695 (POILE) ermittelt für den übergebenen Pointer SPTR die Adresse PTR und Länge LEN jeweils als Doppelwort.

SFB 695



Signal	Bedeutung
SPTR	Referenz der Struktur In diesen Parameter wird die Struktur (und damit die Adresse der Struktur) an den PAD-250e übergeben.
LEN	Ausgang Länge Länge der an SPTR übergebenen Struktur als Doppelwort.
PTR	Ausgang Pointer (PTR) Adresse der an SPTR übergebenen Struktur als Doppelwort.

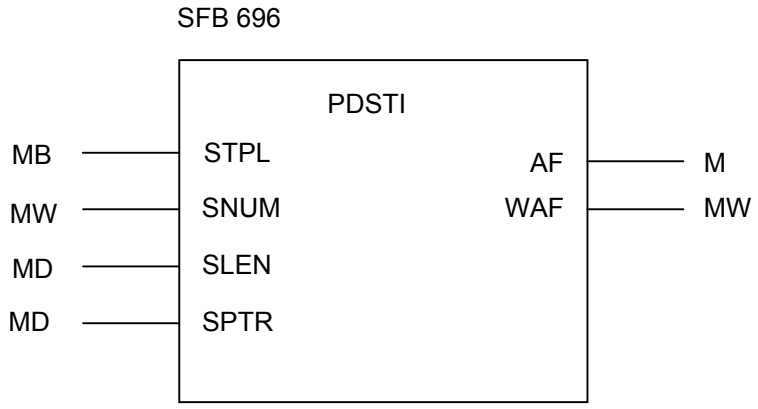
Achtung: Der Baustein liegt ausschließlich als FAR-Version (IM2-Datei) vor. Damit können ausschließlich Strukturen adressiert werden, welche als FAR definiert wurden.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

3.2.4 SFB696 – PDSTI

Der Baustein SFB695 ist Bestandteil des PMCK-Bausteinpaketes und gehört nicht zum Lieferumfang des PDnetIP-Controllers. Das PMCK-Bausteinpaket wurde als Erweiterung zum PDnetIP-Controller PAD-250e zur Kommunikation mit PMCK-Systemen (SICALIS) über das Protokoll RFC1006 entwickelt.

Mit dem SFB696 (PDSTI) wird die Struktur mit der ID SNUM der Adresse SPTR sowie der Länge SLEN an den PAD-250e im Steckplatz STPL übermittelt.



Nach einem Neustart der ALU müssen alle in der Kommunikation genutzten Datenstrukturen registriert werden. Der erste Aufruf muss mit der ID Null erfolgen, um eine Initialisierung der Adresstabelle im PAD-250e zu initiieren.

Eine effektive Variante den Aufruf des SFB696 zu steuern liegt darin, sowohl den Fehlerausgang AF des SFB693 auf eine fallende Flanke als auch den NORM-Merker der SPS auf eine fallende Flanke zu überwachen.

Im Einschaltmoment werden sowohl in der ALU als auch im PAD-250e Initialisierungen durchgeführt. Wann diese abgeschlossen sind, lässt sich im Vorfeld nicht sicher sagen. Daher ist es am einfachsten, den Fehlerausgang AF des SFB693 auf eine fallende Flanke zu überwachen und mit einer Zeitverzögerung die Anmeldung aller Datenstrukturen durchzuführen. Durch diese Aufrufvariante wird sowohl der Einschaltmoment als auch ein Neustart des PDnetIP-Controllers berücksichtigt. Werden die Setupdaten des PAD-250e bei laufender SPS neu geladen, so werden auch die Strukturen automatisch angemeldet.

Achtung: Ab Firmware 4.10.16.18 und Baustein 1.06 beträgt die maximale Laufzeit des Bausteines 20ms.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

3 Installation des PAD-250e
3.2 Sonder-Funktions-Bausteine (SFB)

Signal	Bedeutung
STPL	Steckplatz In dem hier eingetragenen Merkerbyte wird der Steckplatz des PDnetIP-Controllers übergeben. (sinnvolle Werte 2-9)
SNUM	Struktur-Kennung In dem Parameter SNUM wird die Kennung der in der Projektierung von NetPro definierten Struktur übergeben. Die Struktur-Kennung Null löscht alle bisher im PDnetIP-Controller gespeicherten Adressinformationen. Der PDnetIP-Controller arbeitet unabhängig von der ALU der A250. Immer dann, wenn sich die Adresslage von Strukturen in der A250 geändert hat oder die ALU neu gestartet wurde, müssen alle Strukturen bei dem PAD-250e registriert werden.
SLEN	Länge der Struktur welche zuvor über den Baustein SFB695 (POILE) bestimmt wurde.
SPTR	Adresse der Struktur welche zuvor über den Baustein SFB695 (POILE) bestimmt wurde.
AF	Ausgang Fehler Dieser Merker wird dann vom SFB gesetzt, wenn im Laufe der gerade vorangegangenen Bearbeitung ein Fehler aufgetreten ist. Wenn beim nächsten Aufruf kein Fehler festgestellt wird, wird dieser Merker zurückgesetzt. (Ausgang nicht speichernd)
WAF	Wort Ausgang Fehler In diesem Merkerwort legt der SFB einen Fehler-Code ab, der den aufgetretenen Fehler näher beschreibt. Dieses Merkerwort wird vom SFB nie zurückgesetzt (Ausgang speichernd).

Fehlercode	Bedeutung
1	Parameter Steckplatz außerhalb der erlaubten Grenzen
2	Kein PAD-250e mit Strukturinterface im angegebenen Steckplatz gefunden.
110	Die Firmware des PAD-250e unterstützt keine Kommunikation von Datenstrukturen.
111	Ablauffehler der Beauftragung (alte Beauftragung vorhanden)
112	Ablauffehler der Beauftragung (Semaphore ist ungültig)
113	Ablauffehler der Beauftragung (verzögerte Antwort vorhanden)
120	Die Strukturanmeldung wurde vom PAD-250e innerhalb der Wartezeit nicht erfolgreich quittiert.
130	Der Echtzeitkommunikationspuffer für Strukturen ist voll bzw. momentan überlastet.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

3.2.5 SFB691 - PVDMS

In früheren Version der Firmware erfolgte die Kommunikation über den SFB691 (PVDMS) und nicht über den SFB693 (PDNIP). Es existierten zwei Versionen des SFB691. Eine Version zur Kommunikation mit PDnet-Controllern (PAD-250) sowie eine Version zur Kommunikation mit PDnetIP-Controllern (PAD-250e). Dies verhindert den gleichzeitigen Betrieb beider Controller-Typen in einer Steuerung.

Die Aufteilung der Bausteine in zwei getrennte SFB ermöglicht einen gleichzeitigen Betrieb von PDnet- als auch PDnetIP-Controllern in einer Steuerung.

Zur Umstellung eines Programms auf den SFB693 (PDNIP) ist der Aufruf des SFB691 (PVDMS) lediglich zu ersetzen. Die Parameter sind identisch.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

3.2.6 Integration der SFB

Die folgenden Kapitel beschreiben die Integration der Bausteine in AKF125 und ALD25. Das Bausteinpaket sfb_pdn enthält den Baustein SFB691 für den PAD-250 sowie SFB693 für den PAD-250e. Das Bausteinpaket sfb_str enthält den Baustein SFB692 für den PAD-250e. Diese Lösung ermöglicht einen einfachen parallelen Betrieb von PAD-250 sowie PAD-250e in einer Steuerung.

3.2.6.1 AKF125

Die SFB's können in AKF ab Version 4.21 integriert werden. Die folgenden Schritte beziehen sich auf eine Standard-Installation in Laufwerk C, Verzeichnis C:\AEG-A91\AKF125.

Liegt eine andere Installation vor, so sind die Angaben entsprechend anzupassen. Zum Einbinden der mitgelieferten SFB's in AKF125, sind folgende Schritte durchzuführen:

1. AKF aufrufen und eine Anlage und eine Station einstellen z.B. C:\AKF125\BEISPIEL (Anlage=AKF125, Station=BEISPIEL)
2. Alle „.IM0" und „.IM2"-Dateien von der mitgelieferten Diskette in das Verzeichnis C:\AEG-A91\AKF125\SFB25 kopieren. Anschließend in das Verzeichnis C:\ AEG-A91\AKF125 wechseln.
3. AKF wie folgt aufrufen: AKF25 /GETSLIB
   Hierdurch wird AKF veranlasst, die Beschreibungstextdatei zu der momentan bestehenden SFB-Bibliothek zu erzeugen.
   Nach diesem Vorgang befindet sich im Verzeichnis der aktuell eingestellten Station (z.B. C:\AKF125\BEISPIEL) die Datei SFBLIB.GEN.
4. Die Datei SFBLIB.GEN jetzt in einen ASCII Texteditor laden. Zunächst muss die maximale Anzahl der SFB's geändert werden. Dies ist die erste Zahl innerhalb der runden Klammern.

   Beispiel:

   ```
   #GENSFB SFB25.LIB(SFB25_T, 500, 0, 0, 0);
   ```
   ändern in
   ```
   #GENSFB SFB25.LIB(SFB25_T, 699, 0, 0, 0);
   ```

   Gehen Sie anschließend zum Ende der Datei und suchen Sie dort die letzte Zeile, die mit BES beginnt. Fügen Sie nach dieser Zeile zwei neue BES-Zeilen ein. Die Zahl nach dem Schlüsselwort BES ist die Zahl der letzten BES-Zeile + 1. Notieren Sie sich diese Zahl (im Beispiel: 63 und 64)

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

5. **Beispiel:**

```
.....
BES 58 ( 0 ) , ( 0 );
BES 59 ( 0 ) , ( 0 );
BES 60 ( 0 ) , ( 0 );
BES 61 ( 0 ) , ( 0 );
BES 62 ( 0 ) , ( 0 );
```

erweitern in:

```
BES 58 ( 0 ) , ( 0 );
BES 59 ( 0 ) , ( 0 );
BES 60 ( 0 ) , ( 0 );
BES 61 ( 0 ) , ( 0 );
BES 62 ( 0 ) , ( 0 );
BES 63 ( 0 ) , ( 0 );
BES 64 ( 0 ) , ( 0 );
```

6. Jetzt muss die Beschreibungsdatei um die Definitionen für die einzelnen SFB's erweitert werden. Gehen Sie dazu in der Datei in den Bereich vor den BES-Zeilen. Fügen Sie anschließend die Datei APEX.GEN in die Datei ein. Jetzt muss die zuvor in der eingefügten BES-Zeile definierte Nummer (Parameter nach BES) in die neu eingefügte Definition übertragen werden. Die Zahl wird als letzter Parameter in der ersten Zeile jeder Bausteinbeschreibung eingetragen.

Beispiel:

```
SFB 483 (L_AKF, SFB_AKD, 98, 0, 53)
SPFPV (
   WEV          AKDTAB      I F,
   EF           BIT         I ,
   IND          BIT         I ,
   TNR          WORT        I ,
   ATTR         WORT        I ,
   WEK          WORT        I ,
   WEL          WORT        I ,
   TYP          WORT        I ,
   WAB          WORT        0 ,
   WAF          WORT        0
   );

BES 1 ( 0 ) , ( 0 ) ;
BES 2 ( 0 ) , ( 0 ) ;
```

erweitern/ändern in

```
SFB 483 (L_AKF, SFB_AKD, 98, 0, 53)
```


© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

```
SPFPV (
   WEV          AKDTAB      I F,
   EF           BIT         I ,
   IND          BIT         I ,
   TNR          WORT        I ,
   ATTR         WORT        I ,
   WEK          WORT        I ,
   WEL          WORT        I ,
   TYP          WORT        I ,
   WAB          WORT        0 ,
   WAF          WORT        0
   );

SFB 691 ( L_AKF|NOX87, SFB_PDN, 50, 0, 63 )
PVDMS (
 STPL BYTE I,
 NORM BIT I,
 AF BIT O,
 WAF WORT O
 );

SFB692 ( L_AKF|NOX87, SFB_STR, 52, 0, 64 )
PDSTR (
 STPL BYTE I,
 SNUM WORT I,
 SPTR ANY I,
 AF BIT O,
 WAF WORT O
 );

SFB 693 ( L_AKF|NOX87, SFB_PDN, 51, 0, 63 )
PDNIP (
 STPL BYTE I,
 NORM BIT I,
 AF BIT O,
 WAF WORT O
 );

BES 1 ( 0 ) , ( 0 ) ;
BES 2 ( 0 ) , ( 0 ) ;
```

Abschließend die Datei speichern.

7. AKF wie folgt aufrufen: **AKF25 /FWTsfblib.gen**
   Hierdurch wird AKF veranlasst, mit Hilfe der geänderten Beschreibungstextdatei eine neue SFB-Bibliothek zu erzeugen. Nach diesem Vorgang befindet sich im Verzeichnis der aktuell eingestellten Station die Datei PROT.PRT; in der sich eine Liste aller eingebundenen SFB's befindet. Durch Ansehen dieser

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Datei kann man überprüfen, ob alle SFB's eingebunden wurden.

Jetzt können alle mitgelieferten SFB's im AKF-Editor aufgerufen werden.

3.2.6.2 ALD25 für OS/2

Der SFB kann ebenfalls in ALD25 integriert werden. Die Vorgehensweise entspricht der Integration in AKF125 mit den folgenden Abweichungen:

- Das Verzeichnis C:\AEG-A91\AKF125 ist durch C:\MODICON\ALD25 zu ersetzen.
- Der Programmaufruf AKF25 ist durch ALD25 zu ersetzen.

3.2.6.3 ALD25 für Windows NT/XP

Der SFB kann ebenfalls in ALD25 integriert werden. Die Vorgehensweise entspricht der Integration in AKF125 mit den folgenden Abweichungen:

- Das Verzeichnis C:\AEG-A91\AKF125 ist durch C:\MODICON\ALD25 zu ersetzen.
- Der Programmaufruf AKF25 ist durch ALD25Win zu ersetzen.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

3.2.7 Verfügbarkeit

Je nach projektiertem Endgerätetreiber steht die Kommunikation zwischen PDnetIP-Controller und dem Signalspeicher der Steuerung nicht automatisch zur Verfügung. Einige Treiber können erst mit einer initialisierten Remote-Kopplung ihren Betrieb aufnehmen. Um die Remote-Kopplung initialisieren zu können, ist eine geladene ICOS-Projektierung (COMAKF) erforderlich.

Die folgende Tabelle gibt einen Überblick, welcher Treiber sofort oder nach aktivierter Remote-Kopplung zur Verfügung steht.

Sollte der verwendete PDnetIP-Controller mit Paging arbeiten (ab Revision 3, Index 3 möglich) steht die Endgerätekommunikation immer sofort zur Verfügung.

Treiber	Sofort	Remote
DPM ICOS/BIK-114	X	
DPM ICOS/KPH-141		X
DPM-250e (ICOS-Systemkopplung, Datenmodell, SFB693)		X
DPM-250e (Datenmodell, SFB693)	X	

 © 1992-2014 by APEX® automation technologies GmbH, Braunschweig

3.3 AKF/ALD Bestückungsliste

Der PAD-250e erscheint in Abhängigkeit seiner Konfiguration in der Bestückungsliste der SPS. Für die Funktionsfähigkeit der Baugruppe ist eine Eintragung in der Bestückungsliste nicht erforderlich.

Je nach gewählter Konfiguration identifiziert sich der PDnetIP-Controller gegenüber der ALU. Die nachfolgende Tabelle gibt dazu einen Überblick.

Treiber	ID
DPM ICOS/BIK-114	0x45 (BIK-114)
DPM ICOS/KPH-141	0x4E (KPM-111)
DPM-250e (ICOS-Systemkopplung, Datenmodell, SFB63)	0x4E (KPM-111)
DPM-250e (Datenmodell, SFB693)	0x4E (KPM-111)

3.4 Strukturen

Die Kommunikation zwischen Strukturen steht nicht in jedem Treiber zur Verfügung. Folgende Tabelle gibt eine Übersicht, welche Treiber Strukturen unterstützen.

Treiber	Strukturen
DPM ICOS/BIK-114	-
DPM ICOS/KPH-141	X
DPM-250e (ICOS-Systemkopplung, Datenmodell, SFB63)	X
DPM-250e (Datenmodell, SFB693)	X

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

3.5 ICOS Projektierung mit COMAKF

Der PAD-250e wird bei der Nutzung von ICOS-Diensten mit COMAKF projektiert. Dabei sind für den PAD-250e nur wenige Angaben für den Betrieb notwendig. Die folgende Tabelle gibt einen Überblick, welche Kommunikationsparameter aus der Projektierung übernommen werden:

- **Prozedur**
  Die ICOS Prozedur-Kennung entscheidet darüber wie der PDnetIP-Controller einige Informationen interpretiert. Je nach verwendeter Prozedur-Kennung wird als BroadCast-Adresse 127 oder 255 interpretiert.
- **Adress-Byte**
  Es gibt eine Sonderbehandlung des A-Byte, wenn das Protokoll kein A-Byte für die verwendete Prozedur vorsieht. In diesem Fall wird nach einem Regelsatz das A-Byte der Station bestimmt. Wird eine Slave-Prozedur wie BUS_1_SL oder STERN_1_SL projektiert, benutzt der PDnetIP-Controller immer das A-Byte 1. Arbeitet der PDnetIP-Controller mit einer Master-Prozedur (BUS_1_MA, STERN_1_MA), benutzt er immer das A-Byte 240.
- **Sendepuffer+Empfangspuffer**
  Die bei den Original-Baugruppen bestehenden Konventionen zur Größe des Sende- und Empfangspuffers gelten nicht für die PDnetIP-Controller.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

4 Anwendung

Alle folgenden Anmerkungen beziehen sich auf die Konfiguration mit der gelieferten Firmware.

4.1 ICOS/TCP

Um den PAD-250e im Programm der A250 anzusprechen, gibt es keine besonderen Regeln. Durch Setzen der Sendebeauftragungsbits, werden die entsprechenden ICOS-Telegramme übertragen. Es gibt keine Unterschiede im Anwenderprogramm der SPS zwischen einer BIK/KP-Baugruppe und dem PAD-250e.

4.1.1 ICOS-Diagnose Modul

Um die ICOS-Diagnose nutzen zu können, müssen in der Steuerung zwei Strukturen (je 256 x Byte) definiert und über den SFB692 (PDSTR) am PAD-250e angemeldet werden.

- **Struktur 32100**. In dieser Struktur legt der PDnetIP-Controller Informationen zum Verbindungsstatus ab.
- **Struktur 32101**. In dieser Struktur legt der PDnetIP-Controller Informationen zur zuletzt aufgetretenen Störung ab.

Das erste Element der Struktur repräsentiert die Informationen für das A-Byte 1, das zweite Elemente für das A-Byte 2 usw.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Struktur 32100 – Verbindungsstatus

Status	Beschreibung
1	Die Verbindung wurde durch den PDnetIP-Controller erfolgreich aufgebaut.
2	Die Verbindung wurde erfolgreich durch die Partnerstation aufgebaut.
3	Die Verbindung wurde geschlossen.
4	Die Verbindung zur Station ist im Moment gesperrt. Diese Situation tritt ein, wenn der Verbindungsaufbau zur Partnerstation gescheitert war. Der PDnetIP-Controller sperrt die Verbindung zu diesem A-Byte vorübergehend. Damit wird erreicht dass bei fortlaufenden Sendeversuchen nicht der gesamte Sendepuffer blockiert wird.
5	Es wird eine andere Verbindung zu dieser Station genutzt. Dies tritt immer dann auf, wenn unter einer IP-Adresse verschiedene A-Bytes gleichzeitig erreichbar sind. Im allgemeinen sind davon nur PC-Systeme betroffen auf denen gleichzeitig mehrere Applikation laufen, die ICOS/TCP nutzen.
6	Es stehen keine weiteren Sockets zur Verfügung. Der Fehler trat auf, als der PDnetIP-Controller aktiv eine Verbindung zu einer anderen Station aufbauen wollte.
7	Es stehen keine weiteren Sockets zur Verfügung. Der Fehler trat auf als der PDnetIP-Controller eine Verbindung von einer anderen Station annehmen wollte.
8	Der Verbindungsaufbau ist gescheitert.

Struktur 32101 - Fehlerinformationen

Fehler	Beschreibung
1	Auflösung A-Byte in IP-Adresse gescheitert. Dieser Fehler tritt z.B. dann auf, wenn die Partnerstation über einen DNS-Namen erreicht wird und der DNS-Server diesen nicht auflösen konnte.
2	Projektierungsfehler: Das A-Byte ist im Setup des PDnetIP-Controllers nicht definiert. Damit hat der PDnetIP-Controller keine Möglichkeit für das A-Byte eine IP-Adresse zu bestimmen, unter welcher die Station erreichbar ist.
3	Eine Partnerstation mit HotStandBy-Funktionalität hat eine Umschaltung durchgeführt. Damit müssen alle Verbindungen zu diesem A-Byte geschlossen werden.
4	Die Remote-Kopplung wurde durch die lokale Steuerung deaktiviert.
5	Der Verbindungsaufbau zur Partnerstation ist gescheitert.
6	Der IP-Stack meldet für eine Verbindung eine Störung.
7	Die Verbindung wurde durch die Partnerstation geschlossen.
8	Die Verbindung wurde geschlossen nachdem für eine Zeit keine Aktivität mehr vorhanden war
9	Ein nicht bekannter Fehler ist aufgetreten.
100	Nicht mappbarer Fehler des TCP/IP-Stack
101..255	Fehelrcodes des TCP/IP-Stack. Diese sind in einer gesonderten Tabelle im Anhang dokumentiert.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

4.2 ModBus/TCP

Um das Protokoll ModBus/TCP nutzen zu können, muss der Sonderfunktionsbaustein PDNIP (SFB693) zyklisch im Anwenderprogramm der SPS aufgerufen werden. Der Baustein ist für die Kommunikation zwischen dem Signalspeicher der ALU und dem PDnetIP-Controller verantwortlich. Unabhängig von der Art der Anwendung besitzt der Baustein keine treiberspezifischen Parameter.

Soll auf den Signalspeicher der A250 von außen zugegriffen werden, so wird die Server-Funktionalität benötigt. Dazu genügt es, den PDNIP-Baustein zyklisch aufzurufen. Weitere Anpassungen im Anwenderprogramm der SPS sind nicht notwendig.

In der „ModBus-Welt“ wird zwischen den Datentypen Register und Coil unterschieden. Ein Register ist ein Wort (16 Bit) und ein Coil ein Merker. Dieses Handbuch verwendet für lokale Adressangaben im allgemeinen die Namen der SPS-Datentypen. Die Verwendung von Register und Wort sowie von Coil und Merker sind an einigen Stellen entsprechend austauschbar.

Soll die A250 selbst aktiv mit anderen ModBus/TCP-Stationen im Netzwerk kommunizieren (Master-Funktionalität), sind folgende Punkte zu berücksichtigen:

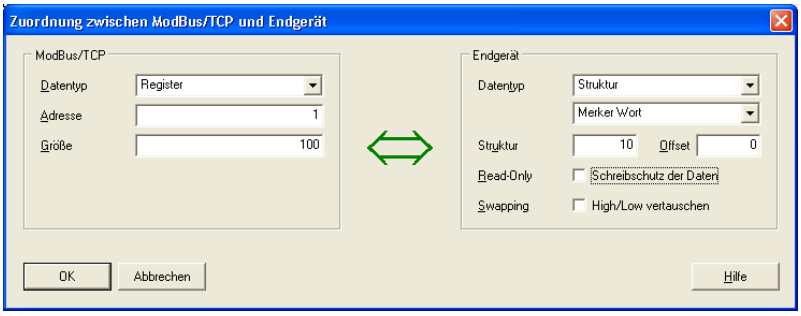
- In der Projektierung müssen ModBus-Auftragsblöcke definiert werden.
- Für jeden ModBus-Auftragsblock ist ein Speicherbereich von 16 Worten in der SPS zu reservieren. Ein ModBus-Auftragsblock darf zwischenzeitlich nicht für andere Aufgaben (Zwischenspeicher) genutzt werden.
- Im Anwenderprogramm der SPS werden die Aufträge über die ModBus-Auftragsblöcke gesteuert.
- Die ModBus-Auftragsblöcke werden nach dem Aufruf des PDNIP-Bausteines vom PDnetIP-Controller ausgewertet und die entsprechenden Aktivitäten ausgelöst. Die Ergebnisse stehen je nach Bearbeitungszeit der Gegenstation entsprechend zeitverzögert im Signalspeicher der SPS zur Verfügung.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

4.2.1 Strukturen

Für die Unterstützung von Strukturen gibt es eine erweiterte Projektierung in NetPro. Beachten Sie die notwendigen Versionen von Firmware, Baustein und NetPro, wie auf Seite 6 beschrieben. Werden andere Versionen genutzt, steht die Kommunikation mit Strukturen nicht zur Verfügung.

Um in NetPro ein Mapping für eine Datenstruktur zu projektieren wird als Datentyp im Endgerät Struktur ausgewählt. Anschließend wird über eine zweite Dropdown-Liste der Datentyp in der Struktur definiert. Die Zuordnung zwischen der eigentlichen Struktur und der Position der Daten innerhalb der Struktur erfolgt über die Kennung als auch dem Offset der Daten innerhalb der Struktur.



Im Anwenderprogramm muss nach einem Neustart des PDnetIP-Controllers jede Struktur mit dem Baustein SFB692 (PDSTR) angemeldet werden (siehe Kapitel 4.2.1).

Ein ModBus-Transfer (Server/Master) muss vollständig von einem Mapping abgedeckt werden. Es gibt keine Möglichkeit Daten aus verschiedenen Mappings gebündelt zu verarbeiten.

Wichtig! Die Firmware prüft nicht, ob die Struktur in der Steuerung eine ausreichende Größe hat. Falsche Projektierungen können daher nicht absehbare Auswirkungen auf die Steuerung haben. Es ist daher auch an dieser Stelle notwendig und wichtig, alle Angaben sorgfältig zu prüfen.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

4.2.2 ModBus-Auftragsblock

Die nachfolgende Tabelle gibt eine Übersicht über alle Elemente eines ModBus-Auftragsblockes.

Wort	Kürzel	Bedeutung
1	Statuswort	Aus dem Statuswort geht der aktuelle Zustand des Auftragsblockes hervor. **1 = IDLE** Der Auftragspuffer ist frei. Dieser Status ist für die Auftragsverwaltung der SPS reserviert und wird vom PDnetIP-Controller nicht benötigt. Der Status muss nicht genutzt werden. **2 = START** Dieser Status übergibt die Kontrolle an den PDnetIP-Controller. Zu diesem Zeitpunkt muss der gesamte Auftragsblock vollständig mit den richtigen Daten gefüllt sein. Das Anwenderprogramm der SPS darf die Werte solange nicht verändern, bis der PDnetIP-Controller die Verarbeitung des Auftragsblockes abgeschlossen hat. Dies wird durch den Status 4 signalisiert. **3 = RUN** Über diesen Status zeigt der PDnetIP-Controller, dass der Auftrag erkannt wurde. Während der gesamten Verarbeitung steht dieser Status an. **4 = DONE** Der PDnetIP-Controller hat den Auftrag abgeschlossen und das Ergebnis des Auftrages kann dem Fehlerwort (Wort 15) entnommen werden. Der Puffer kann jetzt wieder vom Anwenderprogramm der SPS erneut beauftragt werden.
2	Funktionscode	Der hier verwendete Funktionscode wird vom PDnetIP-Controller verarbeitet und entspricht nicht dem MODBUS-Funktionscode. **1 = Read Holding Register** Liest Daten á 16 Bit aus dem 4x-Bereich der Partnerstation. **2 = Write Holding Register** Schreibt Daten á 16 Bit in den 4x-Bereich der Partnerstation. **3 = Read Coil** Liest Daten á 1 Bit aus dem 0x-Bereich der Partnerstation. **4 = Write Coil** Schreibt Daten á 1 Bit in den 0x-Bereich der Partnerstation. **5 = Read Input Discrete** Liest Eingangsdaten á 1 Bit aus dem 1x-Bereich der Partnerstation. **6 = Read Input Register** Liest Eingangsdaten á 16 Bit aus dem 3x-Bereich der Partnerstation.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Wort	Kürzel	Bedeutung
3	Datentyp	Reserviert
4	S-Anzahl	Arbeitet der Block im Scannermode zählt der PDnetIP-Controller in diesem Wort die Anzahl der automatischen Wiederholungen.
5	Lokale Addresse	Adresse des zu transferierenden Signalspeichers in der eigenen (lokalen) Steuerung. Bei Funktionscode 1, 2, 6 ist es eine Register-Adresse und bei Funktionscode 3, 4, 5 eine Coil-Adresse.
6	Anzahl	Anzahl der zu transferierenden Daten. Die Größenangabe erfolgt als Anzahl Elemente, je nach Funktionscode also Anzahl Coils bzw. Anzahl Worte.
7	IP-Adresse 1	IP-Adresse der (anderen) ModBus/TCP Station. Soll die Kommunikation über eine serielle Schnittstelle erfolgen muss für die serielle Schnittstelle 1 der Wert 0.0.0.1 und für die serielle Schnittstelle 2 der Wert 0.0.0.2 eingetragen werden.
8	IP-Adresse 2	
9	IP-Adresse 3	
10	IP-Adresse 4	
11	UnitID	Unit-ID (ModBus/TCP Daten) - Die UnitID wird im ModBus/TCP-Telegramm abgelegt und ist eventuell für die weitere Verarbeitung in der (anderen) ModBus/TCP Station relevant. Dies betrifft jedoch nur Geräte mit Routing-Fähigkeit und muss der jeweiligen Gerätedokumentation entnommen werden.
12	TRID	Transaktions-ID (ModBus/TCP Daten) - Die TRID wird im ModBus/TCP Telegramm abgelegt und hat für die weitere Verarbeitung im PDnetIP-Controller keine Bedeutung. Über die Transaktion-ID kann ein ModBus/TCP Telegramm markiert werden, so dass es sich mit einem Netzwerk-Sniffer eindeutig identifizieren lässt. Dazu ist die Transaktions-ID vor jeder Beauftragung am einfachsten zu inkrementieren.
13	Entfernte Adresse	Registeradresse des anderen (Remote) ModBus/TCP Gerätes. Diese Adresse bestimmt den Speicherbereich (Register/Coils) auf den sich die Operation (Read/Write) bezieht.
14	TimeOut	Ist das Wort mit einem Wert ungleich Null geladen wird der Block im Scannermode bearbeitet. Der PDnetIP-Controller wiederholt den Auftrag automatisch. Der Auftrag wird nur bei einem Fehler oder durch das Löschen des Timeout-Wertes abgebrochen.
15	Fehlerwort	Nach Verarbeitung des Auftrages steht in diesem Wort der Abschlussstatus der Operation. Die Tabelle „Fehlercodes“ dokumentiert die hier möglichen Werte.
16	Info	Steht im Fehlerwort der Fehlercode 112 (Die Partnerstation hat den Auftrag mit einer Exception beantwortet) steht in diesem Wort der in der Exception übertragene „Exception Code“.

Wird vom Anwenderprogramm der SPS ein Auftragsblock gestartet, dürfen keine Änderungen am Auftragsblock durchgeführt werden bis der PDnetIP-Controller im Statuswort den Wert 4 (DONE) ablegt hat. Ein Auftrag wurde erfolgreich durchgeführt, wenn im Fehlerwort 0 steht.

Achtung: ModBus-Auftragsblöcke können nur im Merkerwortbereich der Steuerung arbeiten und somit nicht in Strukturen liegen.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Liste der Fehlercodes in Wort 15

Fehlercode	Bedeutung
0	Kein Fehler
100	Ungültiger Funktionscode im Auftragspuffer übergeben.
101	Es konnte keine Verbindung zur ModBus/TCP Station aufgebaut.
104	Für die im Auftragsblock übergebene lokale Adresse wurde kein Mapping definiert (Funktionscode 1+2: Register, Funktionscode 3+4: Coil)
105	Die übergeben UnitID ist größer als 255
106	Die übergebene entfernte Adresse ist Null
107	Die Anzahl Elemente ist im bei dem gewählten Funktionscode nicht zulässig.
108	Die Kombination aus entfernte Adresse und Anzahl Elemente ist bei dem gewählten Funktionscode nicht zulässig
109	Die Sendedaten konnte an die Partnerstation nicht übermittelt werden. Im allgemeinen wurde die Verbindung zur Partnerstation geschlossen
110	Der Auftrag wurde an die Partnerstation übermittelt und während dem der PDnetIP-Controller auf die Antwort gewartet hat wurde die Verbindung geschlossen. Manche Geräte zeigen durch das Schließen der Verbindung einen Fehlerzustand an.
111	Die empfangene Antwort hat einen falschen Funktionscode.
112	Die Partnerstation hat den Auftrag mit einer Exception beantwortet.
113	Die empfangene Antwort beinhaltet eine andere Transaktions-ID also die welche im Auftrag gesendet wurde.
114	Die Protokoll-ID im Antworttelegramm ist ungültig.
115	Die UnitID im Antworttelegramm ist ungültig.
116	Die Länge des Empfangstelegramms passt nicht zum Auftrag.
117	Die innere und äußere Länge des Empfangstelegramms passen nicht zusammen. Im ModBus-Telegramm ist die Telegrammlänge redundant abgelegt, so dass geprüft werden kann, ob diese zueinander passen.
118	Interner Fehler (omb_send)
119	Der Auftrag konnte nicht rechtzeitig abgeschlossen werden.
120	Das Statuswort wurde normiert, da es vom Anwenderprogramm der SPS in nicht zulässiger Art geändert wurde.
121	Elemente der IP-Adresse im Auftragsblock sind größer als 255
122	Die übergebene IP-Adresse ist ungültig
123	Interner Fehler (VDM_DcToPtr)
124	Die in der Quittung enthaltene Adresse passt nicht zum Auftrag.
125	Die in der Quittung enthaltene Größe passt nicht zum Auftrag.
126	Verbindung wurde durch einen anderen Auftragsblock geschlossen.
127	TRID wird bereits genutzt

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Fehlercode	Bedeutung
128	TRID steht im seriellen ModBus nicht zur Verfügung.
129	UnitId muss im seriellen ModBus gesetzt sein
130	Das Mapping kann für den Request nicht aufgelöst werden.
131	Mapping-Fehler

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

ModBus Exception Codes

Exception-Code	Bedeutung
01 = ILLEGAL FUNCTION	The function code received in the query is not an allowable action for the server (or slave). This may be because the function code is only applicable to newer devices, and was not implemented in the unit selected. It could also indicate that the server (or slave) is in the wrong state to process a request of this type, for example because it is unconfigured and is being asked to return register values.
02 = ILLEGAL DATA ADDRESS	The data address received in the query is not an allowable address for the server (or slave). More specifically, the combination of reference number and transfer length is invalid. For a controller with 100 registers, a request with offset 96 and length 4 would succeed, a request with offset 96 and length 5 will generate exception 02.
03 = ILLEGAL DATA VALUE	A value contained in the query data field is not an allowable value for server (or slave). This indicates a fault in the structure of the remainder of a complex request, such as that the implied length is incorrect. It specifically does NOT mean that a data item submitted for storage in a register has a value outside the expectation of the application program, since the MODBUS protocol is unaware of the significance of any particular value of any particular register.
04 = SLAVE DEVICE FAILURE	An unrecoverable error occurred while the server (or slave) was attempting to perform the requested action.
05 = ACKNOWLEDGE	Specialized use in conjunction with programming commands. The server (or slave) has accepted the request and is processing it, but a long duration of time will be required to do so. This response is returned to prevent a timeout error from occurring in the client (or master). The client (or master) can next issue a Poll Program Complete message to determine if processing is completed.
06 = SLAVE DEVICE BUSY	Specialized use in conjunction with programming commands. The server (or slave) is engaged in processing a long–duration program command. The client (or master) should retransmit the message later when the server (or slave) is free.
08 = MEMORY PARITY ERROR	Specialized use in conjunction with function codes 20 and 21 and reference type 6, to indicate that the extended file area failed to pass a consistency check. The server (or slave) attempted to read record file, but detected a parity error in the memory. The client (or master) can retry the request, but service may be required on the server (or slave) device.
10 (0x0A) = GATEWAY PATH UNAVAILABLE	Specialized use in conjunction with gateways, indicates that the gateway was unable to allocate an internal communication path from the input port to the output port for processing the request. Usually means that the gateway is misconfigured or overloaded.
11 (0x0B) = GATEWAY TARGET DEVICE FAILED TO RESPOND	Specialized use in conjunction with gateways, indicates that no response was obtained from the target device. Usually means that the device is not present on the network.

Quelle: MODBUS Application Protocol Specification V1.1a

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

4.2.3 Beispiele

Nachfolgend werden einige Beispiele für die Anwendung der Master-Funktion gegeben. Die Beispiele beschreiben die Funktionalität und wie der Auftragsblock dazu belegt werden muss. Die im Anwenderprogramm der SPS notwendige Ablaufsteuerung im Statuswort wird nicht beschrieben. Für Testzwecke können die Auftragsblöcke auch über die OnLine-Liste gestartet werden.

4.2.3.1 Read Register

Im Beispiel werden aus der ModBus/TCP Station mit der IP-Adresse 192.168.1.33 aus dem Register 500, 10 Register gelesen und in der lokalen SPS ab Register 2000 abgelegt.

```
W1     Statuswort          2
W2     Funktionscode       1
W3     Datentyp            0
W4     ErwAdresse          0
W5     Lokale Adresse      2000
W6     Anzahl              10
W7     IP-Adresse 1        192
W8     IP-Adresse 2        168
W9     IP-Adresse 3        1
W10    IP-Adresse 4        33
W11    UnitID              0
W12    TRID                0
W13    Entfernte Adresse   500
W14    TimeOut             0
W15    Fehlerwort          0
W16    Info                0
```


© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

4.2.3.2 Write Register

Im Beispiel wird ein Schreibbefehl in die ModBus/TCP Station mit der IP-Adresse 192.168.1.33 ausgelöst. Es werden 25 Merkerworte ab Register 100 in die andere ModBus/TCP-Station ab Register 500 geschrieben.

```
W1     Statuswort          2
W2     Funktionscode       2
W3     Datentyp            0
W4     ErwAdresse          0
W5     Lokale Adresse      100
W6     Anzahl              25
W7     IP-Adresse 1        192
W8     IP-Adresse 2        168
W9     IP-Adresse 3        1
W10    IP-Adresse 4        33
W11    UnitID              0
W12    TRID                0
W13    Entfernte Adresse   500
W14    TimeOut             0
W15    Fehlerwort          0
W16    Info                0
```

4.2.3.3 Read Coil

Im Beispiel werden aus der ModBus/TCP Station mit der IP-Adresse 192.168.1.33 ab dem Coil 800, 10 Coils gelesen und in der lokalen SPS ab Coil 55 abgelegt.

```
W1     Statuswort          2
W2     Funktionscode       3
W3     Datentyp            0
W4     ErwAdresse          0
W5     Lokale Adresse      55
W6     Anzahl              10
W7     IP-Adresse 1        192
W8     IP-Adresse 2        168
W9     IP-Adresse 3        1
W10    IP-Adresse 4        33
W11    UnitID              0
W12    TRID                0
W13    Entfernte Adresse   800
W14    TimeOut             0
W15    Fehlerwort          0
W16    Info                0
```


© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

4.2.3.4 Write Coil

Im Beispiel wird ein Schreibbefehl in die ModBus/TCP Station mit der IP-Adresse 192.168.1.33 ausgelöst. Es werden 30 Merker ab Coil 500 in die andere ModBus/TCP-Station ab Coil 310 geschrieben.

W1	Statuswort	2
W2	Funktionscode	4
W3	Datentyp	0
W4	ErwAdresse	0
W5	Lokale Adresse	500
W6	Anzahl	30
W7	IP-Adresse 1	192
W8	IP-Adresse 2	168
W9	IP-Adresse 3	1
W10	IP-Adresse 4	33
W11	UnitID	0
W12	TRID	0
W13	Entfernte Adresse	310
W14	TimeOut	0
W15	Fehlerwort	0
W16	Info	0

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

4.3 ISO TP0 over TCP (RFC1006)

Um das Protokoll RFC1006 nutzen zu können, muss der Sonderfunktionsbaustein PDNIP (SFB693) zyklisch im Anwenderprogramm der SPS aufgerufen werden. Der Baustein ist für die Kommunikation zwischen dem Signalspeicher der ALU und dem PDnetIP-Controller verantwortlich. Unabhängig von der Art der Anwendung besitzt der Baustein keine treiberspezifischen Parameter.

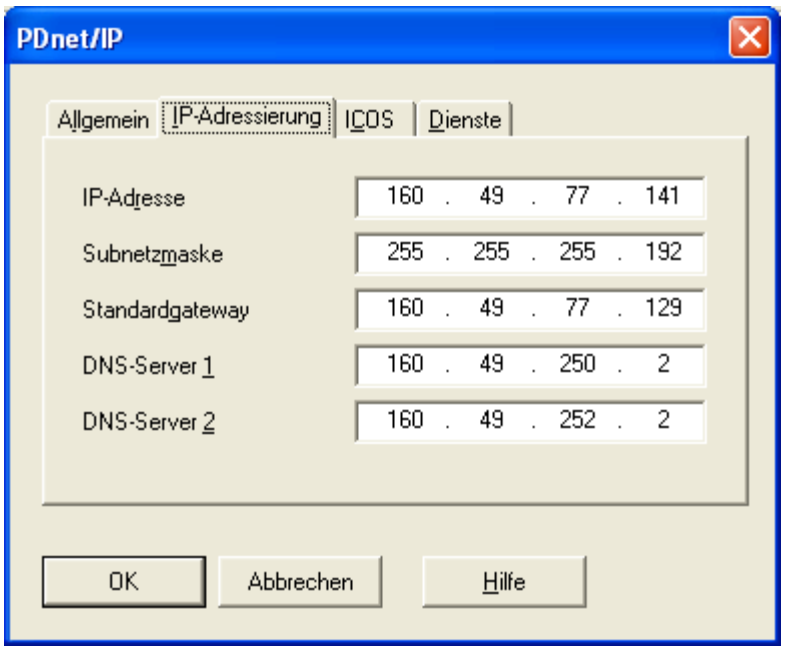
Im Anwenderprogramm der SPS genügt es, den PDNIP-Baustein zyklisch aufzurufen. Weitere Anpassungen im Programm der SPS sind für eine einfache Kommunikation nicht notwendig. Soll die Kommunikation über Strukturen erfolgen, müssen zusätzlich mit dem Baustein PDSTR (SFB692) alle in der RFC-Projektierung genutzten Strukturen beim PDnetIP-Controller registriert werden.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

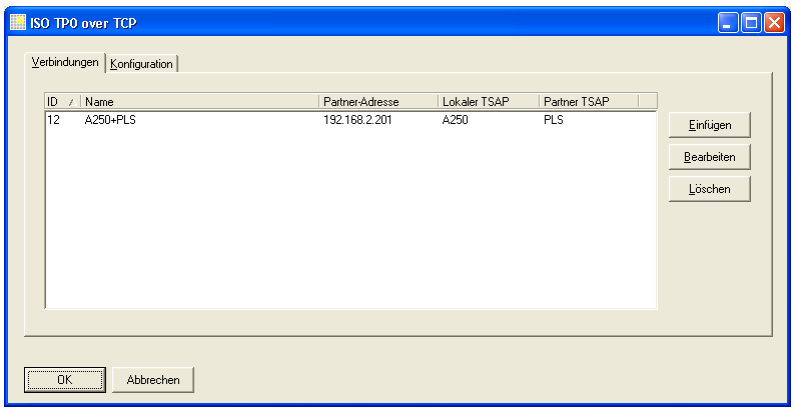
4.3.1 Projektierung mit NetPro

Das Kapitel zeigt Ihnen an einem Beispiel wie eine neue RFC1006 Projektierung erstellt wird. Starten Sie dazu das Projektierungsprogramm NetPro.

Erzeugen Sie ein neues Projekt und fügen Sie im Menüpunkt „Bearbeiten/Station“ einen PAD-250e in das Projekt ein. Wenn Sie den eingefügten PDnetIP-Controller öffnen (Schalter „Bearbeiten“), erreichen Sie unter dem PDnetIP-Kanal den Bearbeiten-Dialog für den PDnetIP-Treiber, in welchen Sie die IP-Adresse und alle weiteren Netzwerk-Parameter definieren.

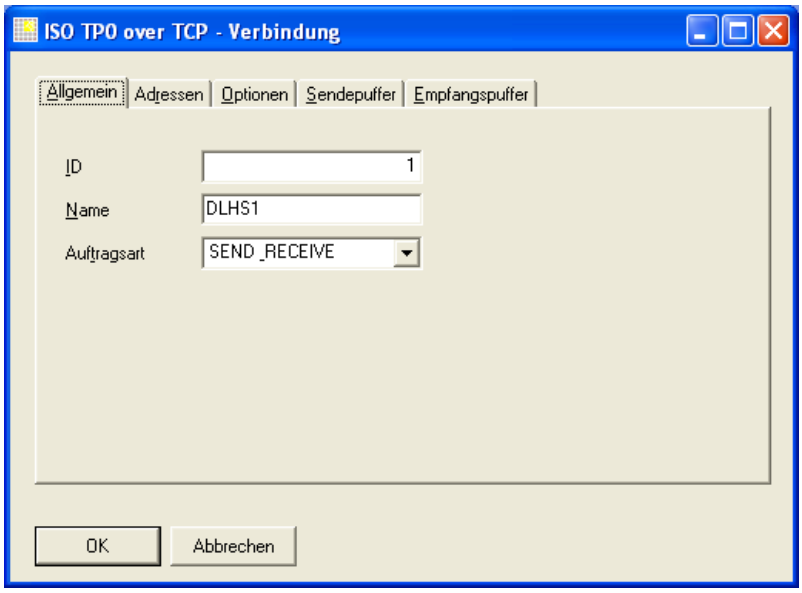


Über die Karteikarte „Dienste“ erreichen Sie die Projektierung der RFC-Option, sowie aller weiteren Netzwerk-Dienste. Die Projektierung der RFC-Option erfolgt als Liste von Verbindungen.



© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

Nachfolgend wird die Bedeutung der einzelnen Informationen beschrieben, welche für jede einzelne Verbindung projektiert werden.



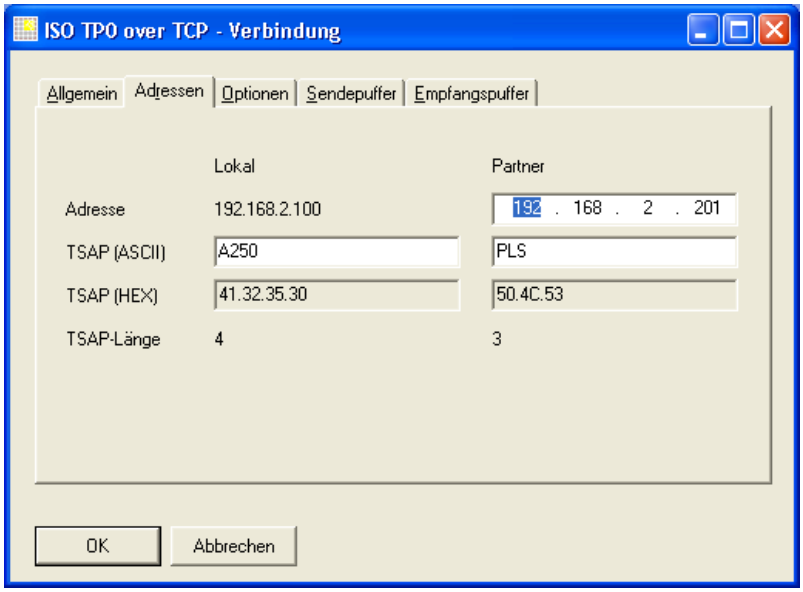
Die Kennung (1..65535), sowie der symbolische Name (24 Zeichen) dienen der eindeutigen Identifikation der Verbindung. Die Identifikation wird vom PDnetIP-Controller ausschließlich intern genutzt. Der Name stellt die Grundlage für die generierten Einträge in der ASD-Datei dar.

Die genutzte Auftragsart (SEND oder RECEIVE) definiert die Richtung der Datenübertragung. Ein Verbindung mit der Auftragsart SEND kann nur Daten senden und keine Daten empfangen. Entsprechend kann eine Verbindung mit der Auftragsart RECEIVE nur Daten empfangen und keine Daten senden. Die Auftragsart SEND/RECEIVE ermöglicht eine bidirektionale Kommunikation.

Wird in der Auftragsart RECEIVE keine IP-Adresse und kein TSAP definiert, wird diese Verbindung als unspezifizierte Verbindung behandelt. Der PDnetIP-Controller akzeptiert dann einen Verbindungsaufbau von einer beliebigen Station.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

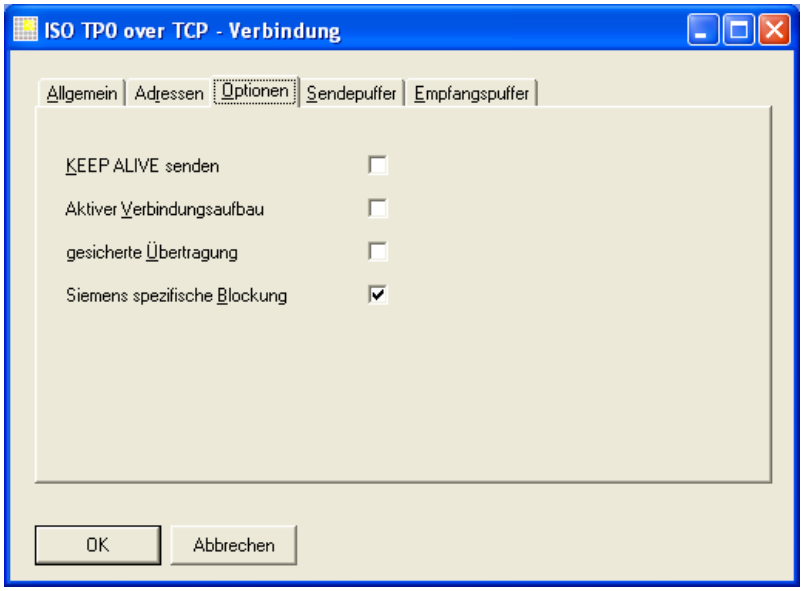
4 Anwendung
4.3 ISO TP0 over TCP (RFC1006)



Die IP-Adresse der Partnerstation identifiziert die Station, mit welcher der PDnetIP-Controller eine Verbindung aufbaut. Wird in den Optionen als Betriebsart RECEIVE gewählt, ist die Eingabe der Partnerstation optional. Der TSAP (Transport Service Access Point) dient der Identifikation der Verbindung auf beiden Seiten (Sender und Empfänger).

Um die Verbindung aufbauen zu können, muss der lokale (eigene) TSAP dem Partner-TSAP, welcher in der Partnerstation definiert ist, entsprechen. Die Länge eines TSAP's beträgt oft 8 Bytes, kann aber im Bereich von 2 Bytes bis 16 Bytes liegen. Der lokale und der Partner-TSAP können identisch sein, da die Verbindung durch die unterschiedlichen IP-Adressen eindeutig ist. Wenn zwei Stationen mehr als eine Verbindung unterhalten sollen, müssen sich auch die TSAP's unterscheiden.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig



Wird die Einstellung „KEEP ALIVE senden“ aktiviert, sendet der PAD-250e zyklisch „KEEP ALIVE“ Telegramme. Ebenfalls wird die Verbindung nicht über ein Inaktivitäts-Timeout geschlossen, sondern nach einem nicht quittierten „KEEP ALIVE“.

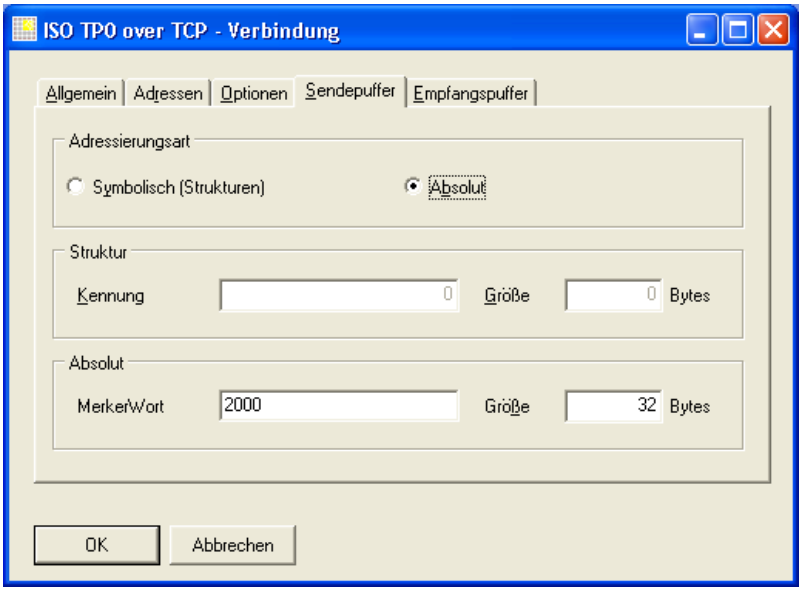
Die Einstellung „Aktiver Verbindungsaufbau“ legt fest, ob die eigene Station aktiv den Verbindungsaufbau betreibt oder passiv den Verbindungsaufbau durch die Partnerstation erwarten soll. Auf beiden Seiten darf nicht die gleiche Einstellung definiert werden.

Der Übertragungsmodus „gesicherte Übertragung“ quittiert von der SPS gesendete Daten zu dem Zeitpunkt wo alle Sendedaten das TCP Output-Window verlassen haben.

In der Übertragungsart „Siemens spezifische Blockung“ sendet der PAD-250e unabhängig von der mit der Partnerstation ausgehandelten TPDU-Größe nie mehr als 400 Byte pro TPDU.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

4 Anwendung
4.3 ISO TP0 over TCP (RFC1006)



Die Adresslage der Sende-/Empfangspuffer definiert den Speicherbereich, in dem der PDnetIP-Controller die Empfangsdaten ablegt oder die Sendedaten entnimmt. Die Adresslage wird dabei für Strukturen als Name (symbolisch) und für andere Adressbereiche absolut als Signalspeicheradresse definiert.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

4.4 OpenTCP-Schnittstelle

Die OpenTCP-Schnittstelle ermöglicht dem Anwenderprogramm der SPS einen direkten Zugriff auf die TCP-Nutzdaten. Um die OpenTCP-Schnittstelle nutzen zu können, muss der Sonderfunktionsbaustein PDNIP (SFB693) zyklisch im Anwenderprogramm der SPS aufgerufen werden. Weitere Anpassungen im Anwenderprogramm der SPS sind für eine einfache Kommunikation nicht notwendig. Soll die Kommunikation über Strukturen erfolgen, müssen zusätzlich mit dem Baustein PDSTR (SFB692) im NORM-Zyklus alle in der TCP-Projektierung genutzten Strukturen beim PDnetIP-Controller registriert werden.

Zur Integration der OpenTCP-Schnittstelle sind folgende Punkte zu berücksichtigen:

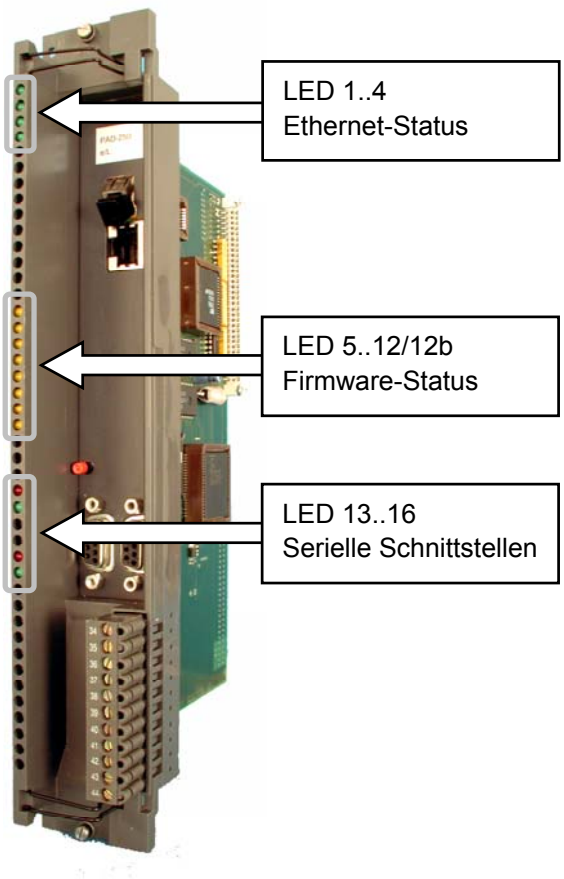
- Die TCP-Verbindungen sind mit NetPro zu projektieren und in den PDnetIP-Controller zu laden.
- Im Anwenderprogramm der SPS ist der PDNIP-Baustein zyklisch aufzurufen. Zur Beschleunigung der Abarbeitung kann der Baustein auch mehrfach hintereinander aufgerufen werden. Dies ist dann sinnvoll wenn die ALU eine große Zykluszeit aufweist.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

5 Anzeige- und Bedienelemente

5.1 Status-Leuchtdioden

Zur Diagnose ohne spezielle Hilfsmittel besitzt der PDnetIP-Controller 16 Leuchtdioden (LED) zur Darstellung des aktuellen Betriebszustandes.



© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

5.1.1 Ethernet Status

Die vier grünen Leuchtdioden 1 bis 4 dienen der Diagnose des Hardware-Status der Ethernet-Schnittstelle. Die Steuerung der Leuchtdioden erfolgt direkt von der Hardware und ist von der Firmware unabhängig. Erfolgt die Verbindung über die RJ45-Schnittstelle, zeigen die Leuchtdioden den aktuellen Status an. Je nach Verbindungsart wechselt die Anzeige.

Erfolgt die Verbindung über den MT-RJ-Anschluss (PAD-250e/L), zeigen die Leichtdioden die eingestellte Betriebsart des PDnetIP-Controllers an. Dies geschieht unabhängig von der Betriebsart der Gegenstelle.

Nr.	LED	Zustand	Bedeutung
1	100 MBit grün	an	100-Mbit Verbindung
		aus	10-Mbit Verbindung
2	FDX grün	an	Voll-Duplex Verbindung
		aus	Halb-Duplex Verbindung
3	ACT grün	flackert	PDnetIP-Controller empfängt Daten
		aus	Keine Datenübertragung
4	LINK grün	an	Verbindung vorhanden
		aus	Keine Verbindung vorhanden

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

5.1.2 Firmware Status

Die Leuchtdioden 5 bis 12 bzw. 12b dienen der Diagnose des Firmware-Status und werden ausschließlich durch die Firmware gesteuert. Unter der Leuchtdiode 12 gibt es optional eine blaue Leuchtdiode 12b. Die Bedeutung der einzelnen Leuchtdioden ist teilweise abhängig von der in den PDnetIP-Controller geladenen Projektierung. In weiteren Beschreibungen von Firmware-Modulen wird im allgemeinen dieser Block herangezogen.

Nr.	LED	Zustand	Bedeutung
5	PAD-Sts. gelb	An	Normalzustand, kein Fehler
		Blinkt	Firmwaremodul ohne spezielle Diagnose LED meldet Fehler (z.B. nicht registrierte Struktur soll übertragen werden).
6	Hardware gelb	Aus	Normalzustand, kein Fehler
		Blinkt	Firmware hat Hardware Fehler erkannt
7	Setup gelb	An	PDnetIP-Controller besitzt gültige Setupdaten
		Blinkt	Setupdaten fehlerhaft nicht vorhanden oder Firmware und Setupdaten passen nicht zueinander oder der PDnetIP-Controller besitzt nicht die erforderliche Freischaltung um ein Modul nutzen zu können (ICOS, RFC1006, freies TCP-Interface).
8	Remote PG gelb	Aus	Normaler Betriebszustand
		an	Der PDnetIP-Controller befindet sich im Lade-Modus.
		blinkt	Es trat ein Fehler in dieser Verbindung auf
9	Bus Status gelb	an	LAN Kanal arbeitet fehlerfrei
		blinkt	LAN Kanal gestört, kein Link am Ethernet-Port vorhanden
10	Ser.1 Status gelb	aus	Es wurde kein Treiber für die serielle Schnittstelle geladen
		an	Es wurde ein Treiber für die serielle Schnittstelle geladen
		blinkt	Fehler in der Kommunikation der seriellen Schnittstelle
11	Ser.2 Status gelb	aus	Es wurde kein Treiber für die serielle Schnittstelle geladen.
		an	Es wurde ein Treiber für die serielle Schnittstelle geladen.
		blinkt	Fehler in der Kommunikation der seriellen Schnittstelle
12	DPM Status gelb	an	Die Kommunikation mit der SPS arbeitet fehlerfrei.
		aus	Es wurde kein Endgerätetreiber zur Kommunikation mit der SPS aktiviert.
		blinkt	Die Verbindung zwischen PDnetIP-Controller und SPS war oder ist gestört. Dies kann die Kommunikation des Datenmodells (SFB691/SFB693) oder die ICOS-Kommunikation betreffen.
12 b	ICOS Status blau	an	Der PDnetIP-Controller verfügt über eine erfolgreich initialisierte ICOS-Kommunikation.
		aus	Keine ICOS-Kommunikation initialisiert.

Achtung: Blinkt die LED 5 (PAD-Sts) gleichzeitig mit LED 6 (Hardware) und die LED 7..12 sind dunkel, so ist diese Firmware auf diesem PDnetIP-Controller nicht einsetzbar.

 © 1992-2014 by APEX® automation technologies GmbH, Braunschweig

5.1.3 Serielle Schnittstellen

Die Leuchtdioden 13 bis 16 dienen der Diagnose der seriellen Schnittstellen. Die Steuerung der Leuchtdioden erfolgt direkt von der Hardware und ist von der Firmware unabhängig. Der Datenempfang der seriellen Schnittstellen kann durch ein Aufleuchten der entsprechenden Empfangs-LED der seriellen Schnittstelle geprüft werden.

Nr.	LED	Funktion	Bedeutung
13	Ser.1 TD Rot	an	Es werden Daten gesendet
		aus	Es werden keine Daten gesendet
14	Ser.1 RD grün	an	Es werden Daten empfangen
		aus	Es werden keine Daten empfangen
15	Ser.2 TD Rot	an	Es werden Daten gesendet
		aus	Es werden keine Daten gesendet
16	Ser.2 RD grün	an	Es werden Daten empfangen
		aus	Es werden keine Daten empfangen

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

5.2 Taster

Je nach Betriebszustand der PDnetIP-Controller ist entweder eine Firmware oder der Lader aktiv. Ob der PDnetIP-Controller sich im Lader befindet, wird über die Leuchtdiode Remote-PG unterschieden. Ist die Leuchtdiode Remote-PG an oder blinkt, so ist der Lader aktiv andernfalls die Firmware.

Funktionen des Tasters bei aktiver Firmware

Funktion	Bedeutung
Fehler quittieren	Taster kurz drücken (ca. 1 Sek.) Die internen Fehlermerker der Firmware werden gelöscht. Blinkende LED's, deren Fehlerursache nicht mehr besteht, leuchten wieder dauernd.
Lader aktivieren	Der Taster wird gedrückt gehalten und die gelben LED's gehen der Reihe nach an, bis alle acht LED´s an sind. Die Firmware führt anschließend einen Restart aus und startet den Lader. Damit der Lader die Firmware nicht automatisch startet, muss der Taster gedrückt bleiben, bis alle gelben LED erneut an sind. Der PDnetIP-Controller befindet sich nun im Lader, die Leuchtdioden Seriell1, Seriell2 sind an und Remote-PG blinkt, alle anderen gelben Leuchtdioden sind aus.

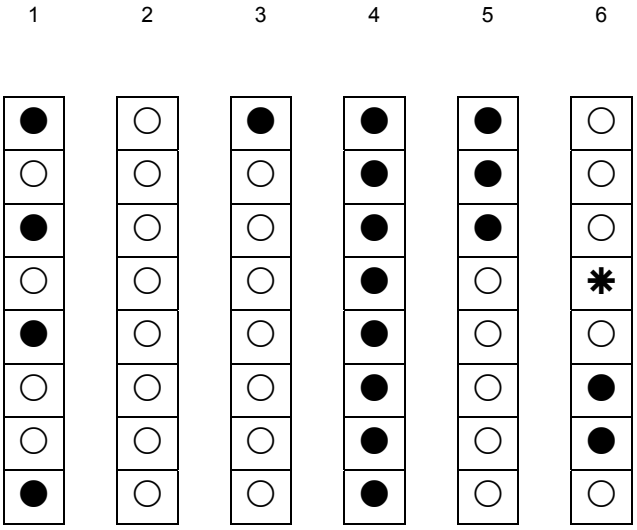
Funktionen des Tasters bei aktivem Lader

Funktion	Bedeutung
Lader verlassen	Der Taster wird gedrückt gehalten und die gelben LED's gehen der Reihe nach an, bis alle acht LED´s an sind. Der Taster muss jetzt losgelassen werden. Der Lader führt anschließend einen Restart aus und startet den PDnetIP-Controller neu.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

5.2.1 Lader aktivieren

Die folgende Tabelle zeigt wie der Lader aktiviert wird wenn eine Firmware läuft. In jeder Spalte ist das Abbild der gelben Leuchtdioden 5..12 markiert.

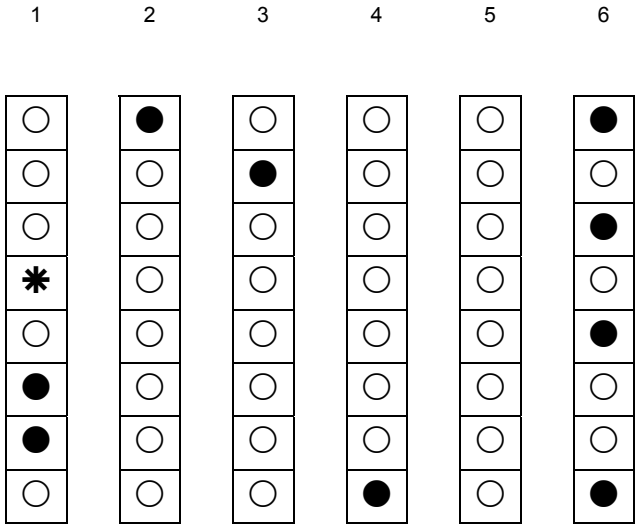


Spalte	Bedeutung
1	Es läuft eine Firmware und es stehen keine Fehler an.
2	Der Taster wurde gedrückt und es gehen alle Leuchtdioden aus.
3	Im Sekundentakt gehen weitere Leuchtdioden an.
4	Nach acht Sekunden sind alle Leuchtdioden an, damit wurde die Firmware informiert dass ein Neustart durchgeführt werden soll.
5	Der PDnetIP-Controller führt einen Neustart durch und es gehen wiederum alle Leuchtdioden der Reihe nach an
6	Anschließend befindet sich der PDnetIP-Controller im Lader, was durch das Blinken der Remote-PG Leuchtdiode erkennbar ist.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

5.2.2 Lader verlassen

Die folgende Tabelle zeigt wie ein aktivierter Lader wieder verlassen werden kann.



Spalte	Bedeutung
1	Es läuft eine Lader.
2	Der Taster wurde gedrückt und es gehen alle Leuchtdioden aus.
3	Etwas schneller als im Sekundentakt wandert die Leuchtdiode nach „unten“.
4	Alle gelben Leuchtdioden wurden durchlaufen..
5	Wenn die letzte gelbe Leuchtdiode erlischt führt der PDnetIP-Controller einen Neustart durch. In diesem Moment muss der Taster losgelassen werden, damit die Firmware starten kann.
6	Anschließend läuft im PDnetIP-Controller die Firmware.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

6 Konfiguration

6.1 Projektierung

Die Projektierung der PDnetIP-Controller erfolgt vollständig mit NetPro. Jede Konfigurationsänderung wird mit NetPro über die serielle Schnittstelle in den PDnetIP-Controller geladen.

Die Setupdaten und Firmware werden voneinander unabhängig im PDnetIP-Controller gespeichert. Wird eine neue Firmware geladen, bleiben die Setupdaten erhalten. Ein erneuter Programmiervorgang ist nicht notwendig.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

6.2 Firmware

Die Firmware des PDnetIP-Controllers wird mit dem Programm DLSE32 in den PDnetIP-Controller geladen. Das Programm kann unter den aktuellen Windows-Versionen (95,98,ME,NT,XP) eingesetzt werden.

Das Laden einer neuen Konfiguration inkl. Firmware erfolgt in der nachfolgend beschriebenen Reihenfolge.

- Lademodus des PDnetIP-Controllers aktivieren
- Serielle Verbindung zwischen PC und PDnetIP-Controller herstellen (YDL52).
- DLSE32 starten
- Firmware über den Menüpunkt "Firmware"/"Auswählen" selektieren.
- Download der Firmware über den Menüpunkt OnLine/Download.
- Nach den Download führt der PDnetIP-Controller einen Neustart durch.
- Anschließend sind die Setupdaten mit NetPro zu projektieren und in den PDnetIP-Controller zu laden. Dazu muss dieser erneut in den Ladezustand versetzt werden.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

6.3 Lader

Der Lader befindet sich wie die Firmware und die Setupdaten auf dem PDnetIP-Controller im Flash-Memory. Jeder PDnetIP-Controller wird mit einem Lader ausgeliefert und ist ohne Lader nicht funktionsfähig.

In den Firmware-Versionen 1.xx sowie 2.xx sind die Setupdaten direkt in das Firmware-Image integriert. Ab Firmware-Version 3.xx ist die Firmware von dem Setup getrennt. Um eine Firmware ab Version 3.xx einsetzen zu können, wird dazu (minimal) die Lader-Version 2 benötigt. Die aktuelle Lader-Version 5 wird empfohlen, da diese auch große Firmware-Versionen laden kann.

Welche Lader-Version eingesetzt wird, können Sie mit DLSE/32 bestimmen. Starten Sie dazu den PDnetIP-Controller im Lader und führen Sie die Funktion Online/Info aus.

Sollten Sie über einen älteren Lader verfügen und eine neue Version benötigen, so muss der PDnetIP-Controller zur Aktualisierung des Laders eingesandt werden.

Alternativ kann der Lader mit einer Spezial-Firmware aktualisiert werden. Der Einsatz dieser Firmware erfolgt auf eigenes Risiko. Eine Anwendung ohne Ersatzkarte ist in jedem Fall zu vermeiden. Scheitert die Aktualisierung ist der PDnetIP-Controller nicht mehr funktionsfähig! Zur Reaktivierung muss der PDnetIP-Controller eingesandt werden.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

6.3.1 Lader-Versionen

Die folgende Tabelle gibt eine Übersicht über die Baudrate, mit welcher der Lader über die serielle Schnittstelle kommuniziert. Die genutzte Baudrate muss bei der Konfiguration von DLSE/32 bzw. NetPro beachtet werden.

Lader-Version	COM1	COM2
1	19.200 Baud	19.200 Baud
2		115.200 Baud
3		
4		
5		
6	115.200 Baud	
7		
8		

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

6.3.2 Lader-Aktualisierung

Die folgende Anleitung beschreibt die einzelnen Punkte zur Lader-Aktualisierung.

Beachten Sie die zuvor gemachten Hinweise zum möglichen Risiko einer Lader-Aktualisierung!

Ablauf der Lader-Aktualisierung:

1. Bringen Sie den PDnetIP-Controller mit Hilfe des roten Tasters in den Lader.
2. Laden Sie mit DLSE/32 die Spezial-Firmware mit dem Dateinamen „**`Lader-Update auf Version x-yy.bin`**“ in den PDnetIP-Controller.
3. Anschließend führt der PDnetIP-Controller einen Neustart durch und signalisiert über die Leuchtdioden 5..12 (gelber Block) den Programmiervorgang.
4. Der Abschluss der Programmierung wird durch das leuchten der Leuchtdiode 8 angezeigt. Drücken Sie nun zum aktivieren des neuen Laders den Taster bis die Leuchtdioden 5..12 (gelber Block) alle der Reihe nach eingeschaltet wurden.
5. Laden Sie jetzt eine Firmware in den PDnetIP-Controller. Mit dem Firmware-Download ist die Lader-Aktualisierung abgeschlossen.

.

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

7 Anhang

7.1 Fehlerliste TCP/IP-Stack

Die hier aufgeführten Fehlermeldungen sind bei der Diagnose hilfreich. Einige Firmware-Module (ICOS-Diagnose, Ausgaben der Diagnose-Firmware) protokollieren unter Umständen Fehlercodes des TCP/IP-Stack.

Fehler	Beschreibung
101	endpoint address not available
102	address in use
103	family not supported
104	arp table full
105	invalid baud rate
106	invalid comm port number
107	invalid device type
108	invalid interface number
109	invalid mask (ether must not be all fs)
110	invalid ping response
111	endpoint refused connection
112	destination address is required
113	destination unreachable (ICMP)
114	invalid parameter (pointer is 0)
115	interface closed
116	interface table full
117	interface open failed
118	operation (connect) is in progress
119	invalid function call (parameter)

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig

7 Anhang
7.1 Fehlerliste TCP/IP-Stack

Fehler	Beschreibung
120	socket is already connected
121	multicast table full
122	multicast address not found
123	out of ports
124	network is down (send failed)
125	network unreachable (keepalive failed)
126	out of DCUs (packets)
127	option parameter is invalid
128	socket is not connected
129	RTIP not initialized
130	invalid socket descriptor
131	not enough devices
132	socket type or specified operation not
133	send failed due to output list
134	could not determine device
135	non-reentrancy error
136	routing table entry not found
137	routing table full
138	resource initialization failed
139	illegal operation due to socket shutdown
140	Timeout
141	type not supported
142	send needs to ARP but ARP is disabled
143	not enough heap memory for allocate request
144	table full (such as ARP cache, device table etc)
145	Illegal sized packet
146	device open failed
200	The global host table is full
201	The host table entry wasn't found
202	To many dns-servers
203	DNS server returned error
204	DNS name is too long
205	DNS call to socket function failed
206	data not available
207	socket call within DNS failed; probably No response received from server
208	No server in server table

© 1992-2014 by APEX® automation technologies GmbH, Braunschweig